COMPUTER SYSTEMS LABORATORY

STANFORD UNIVERSITY · STANFORD, CA 94305-2192

OK - DTIC

COMPARATIVE PERFORMANCE OF BROADCAST BUS LOCAL AREA NETWORKS WITH VOICE AND DATA TRAFFIC

Timothy A. Gonsalves

Technical Report: CSL-TR-87-317

March 1987

This report is the author's Ph.D. dissertation which was completed under the advisorship of Professor Fouad A. Tobagi. This work was supported by the Defense Advanced Research Projects Agency under Contract No. MDA 903-84-K-0249 and an IBM Graduate Fellowship.

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER 87-317	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER

4. TITLE (and Subtitle) COMPARATIVE PERFORMANCE OF BROADCAST BUS LOCAL AREA NETWORKS WITH VOICE AND DATA TRAFFIC	5. TYPE OF REPORT & PERIOD COVERED TECHNICAL REPORT 6. PERFORMING ORG. REPORT NUMBER 87-317
7. AUTHOR(s) Timothy A. Gonsalves	8. CONTRACT OR GRANT NUMBER(s) MDA 903-84-K-0249
9. PERFORMING ORGANIZATION NAME AND ADDRESS Stanford Electronics Laboratory Stanford University Stanford, CA 94305-2192	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Defense Advanced Research Projects Agency Information Processing Techniques Office 1400 Wilson Blvd., Arlington, VA 22209	12. REPORT DATE March 1987 13. NUMBER OF PAGES 214
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office) Resident Representative Office of Naval Research Durand 165 Stanford University, Stanford, CA 94305-2192	15. SECURITY CLASS. (of this report) UNCLASSIFIED 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Recently, local area networks have come into widespread use for computer communications. Together with the trend towards digital transmission of telephone signals, this has sparked interest in the use of computer networks for the transmission of integrated voice/data traffic. This work addresses two related aspects of local area network performance, a detailed characterization of the performance of Carrier Sense Multiple Access with Collision Detection (CSMA/CD), and the comparative performance of several broadcast bus networks with voice/data traffic.

While prior analysis of CSMA/CD has shown that the protocol achieves good performance with data traffic over a range of conditions, the widely used IEEE 802.3 (Ethernet) implementation of the protocol has several aspects that are not easily amenable to mathematical analysis. These include the binary-exponential back-off algorithm used upon collision, the number of buffers per station, and the physical distribution of stations. Performance measurements on operational 3 and 10 Mb/s networks are presented. These demonstrate that the protocol achieves high throughput with data traffic when the packet transmission time is long compared to the propagation delay, as predicted by analysis. However, at 10 Mb/s, with short packets on the order of 64 bytes, performance is poorer. The inflexibility of measurement leads to the use of simulation to further study the behaviour of the Ethernet. It is shown that, with large numbers of stations, while the throughput of the standard Ethernet is poor, a simple modification to the retransmission algorithm enables near-optimal throughput to be achieved. The effects of the number of buffers and of various distributions of stations are quantified. It is shown that stations near the ends of the network and isolated stations achieve lower than average performance. (SHL)

The second focus of this research is the performance of broadcast bus networks with integrated voice/data traffic. The networks considered are the contention-based Ethernet and two contention-free round-robin schemes, Expressnet and the IEEE 802.4 Token Bus. To accommodate voice traffic on such networks, a new variable-length voice packetization scheme is proposed which achieves high efficiency at high loads. While several studies of voice/data traffic on local area networks have appeared in the literature, the differing assumptions and performance metrics used render comparisons with one another difficult. For consistency, a network-independent framework for evaluation of voice/data networks is formulated. Using simulation, a systematic evaluation is undertaken to determine the regions of good performance of the networks under consideration. Interactions between the traffic types and protocol features are studied. It is shown that the deterministic schemes almost always perform better than the contention scheme. Two priority mechanisms for voice/data traffic on round-robin networks are investigated. These are alternating round mechanism and the token rotation times mechanism which restricts access rights based on the time taken for a token to make one round. An important aspect of this work is the accurate characterization of performance over a wide region of the design space of voice/data networks.

COMPARATIVE PERFORMANCE OF BROADCAST BUS LOCAL AREA NETWORKS WITH VOICE AND DATA TRAFFIC

Timothy A. Gonsalves

Technical Report: CSL-TR-87-317

March 1987

Computer Systems Laboratory
Departments of Electrical Engineering and Computer Science
Stanford University
Stanford, California 94305-2191

Abstract

Recently, local area networks have come into widespread use for computer communications. Together with the trend towards digital transmission of telephone signals, this has sparked interest in the use of computer networks for the transmission of integrated voice/data traffic. This work addresses two related aspects of local area network performance, a detailed characterization of the performance of Carrier Sense Multiple Access with Collision Detection (CSMA/CD), and the comparative performance of several broadcast bus networks with voice/data traffic.

While prior analysis of CSMA/CD has shown that the protocol achieves good performance with data traffic over a range of conditions, the widely used IEEE 802.3 (Ethernet) implementation of the protocol has several aspects that are not easily amenable to mathematical analysis. These include the binary-exponential back-off algorithm used upon collision, the number of buffers per station, and the physical distribution of stations. Performance measurements on operational 3 and 10 Mb/s networks are presented. These demonstrate that the protocol achieves high throughput with data traffic when the packet transmission time is long compared to the propagation delay, as predicted by analysis. However, at 10 Mb/s, with short packets on the order of 64 bytes, performance is poorer. The inflexibility of measurement leads to the use of simulation to further study the behaviour of the Ethernet. It is shown that, with large numbers of stations, while the throughput of the standard Ethernet is poor, a simple modification to the retransmission algorithm enables near-optimal throughput to be achieved. The effects of the number of buffers and of various distributions of stations are quantified. It is shown that stations near the ends of the network and isolated stations achieve lower than average performance.

The second focus of this research is the performance of broadcast bus networks with integrated voice/data traffic. The networks considered are the contention-based Ethernet and two contention-free round-robin schemes, Expressnet and the IEEE 802.4 Token Bus. To accommodate voice traffic on such networks, a new variable-length voice packetization scheme is proposed which achieves high efficiency at high loads. While several studies of voice/data traffic on local area networks have appeared in the literature, the differing assumptions and performance metrics used render comparisons with one another difficult. For consistency, a network-independent framework for evaluation of voice/data networks is formulated. Using simulation, a systematic evaluation is undertaken to determine the regions of good performance of the networks under consideration. Interactions between the traffic types and protocol features are studied. It is shown that the deterministic schemes almost always perform better than the contention scheme. Two priority mechanisms for voice/data traffic on round-robin networks are investigated. These are alternating round mechanism and the token rotation times mechanism which restricts access rights based on the time taken for a token to make one round. An important aspect of this work is the accurate characterization of performance over a wide region of the design space of voice/data networks.

Acknowledgements

Many people have participated, knowingly or otherwise, in this endeavour. Several colleagues at the Computer Systems Laboratory and at Xerox helped in various ways. I would like to thank B. Kumar, Forest Baskett and Yogen Dalal for guidance in the early stages of this work. David Shur gave freely of his time during many discussions that clarified my thinking at sticky points. The presentation of this thesis owes much to my readers, Mike Flynn and Thomas Kailath. To Foaud Tobagi, my thanks for working closely and critically with me through to the end. To Jill Sigl, my appreciation for guiding me through a maze of administrative detail. I am grateful to IBM's Watson Research Center for supporting me with a fellowship for 3 years, and to the Xerox Palo Alto Research Centers for providing experimental facilities.

My stay at Stanford was greatly enriched by friends too numerous to name individually. Those who endured the most for the longest are Vikas Sonwalkar and Mark Chesters. Special mention must be made of Donna Bolster who showed me the rich expanse beyond the land of computers and terminals, and of Loretta Collier who indeed provided a home away from home. The staff and "regulars" of the I-Center deserve credit for creating an atmosphere so congenial that one leaves with reluctance.

To Prilla, who shared it all, my thanks for demonstrating that it could be done, for constant encouragement, and for unfailing patience in the face of many missed deadlines. Littlest but not least was Danica Anjali Maria who, by her arrival on the scene 17 months ago, provided the incentive to complete this work. Her cheerful disregard of the serious and weighty and her irrepressible curiosity provided welcome relief from the rigours of the past months.

To the memory of my mother, Rani Gonsalves, I dedicate this thesis.

Table of Contents

List of Figures

List of Tables

Chapter 1
Introduction

1.1. Historical Perspective

The late 1960's saw the emergence of packet-switching as a means of efficiently sharing expensive communication networks among many users [Roberts 78]. This technique is useful when the traffic is *bursty*, i.e., the traffic consists of messages separated by idle periods of variable duration. In packet-switched networks, rather than dedicating a circuit to a pair of users for the duration of a session, messages are transmitted in separate packets with communication lines being utilized by a pair of users only for the duration of each packet transmission. Long messages may be broken up into sub-units with each being transmitted in a separate packet. Each packet may traverse the path from the source to the destination in several hops, with storage at intermediate nodes in *store-and-forward* networks, leading to problems of routing and ordering of packets at the destination.

In the early 1970's, *local area networks* were introduced for interconnection of computers within an area such as a campus or building [Clark *et. al.* 78]. Such networks are characterized by high bandwidths, currently on the order of 1 - 100 Mb/s, and span distances of 1 - 10 km. Local area networks may be broadly classified according to topology into star, ring and bus networks (Figure 1-1). Star networks are well suited for connection of terminals to a time-shared computer. For the interconnection of autonomous computers they have the drawback of having a single point of failure that completely disrupts operation.

The ring and bus topologies are more suitable for computer interconnections. In the bus, every station receives every packet resulting in *broadcast* operation and eliminating the problems of routing and ordering of packets. With rings, broadcast operation can be

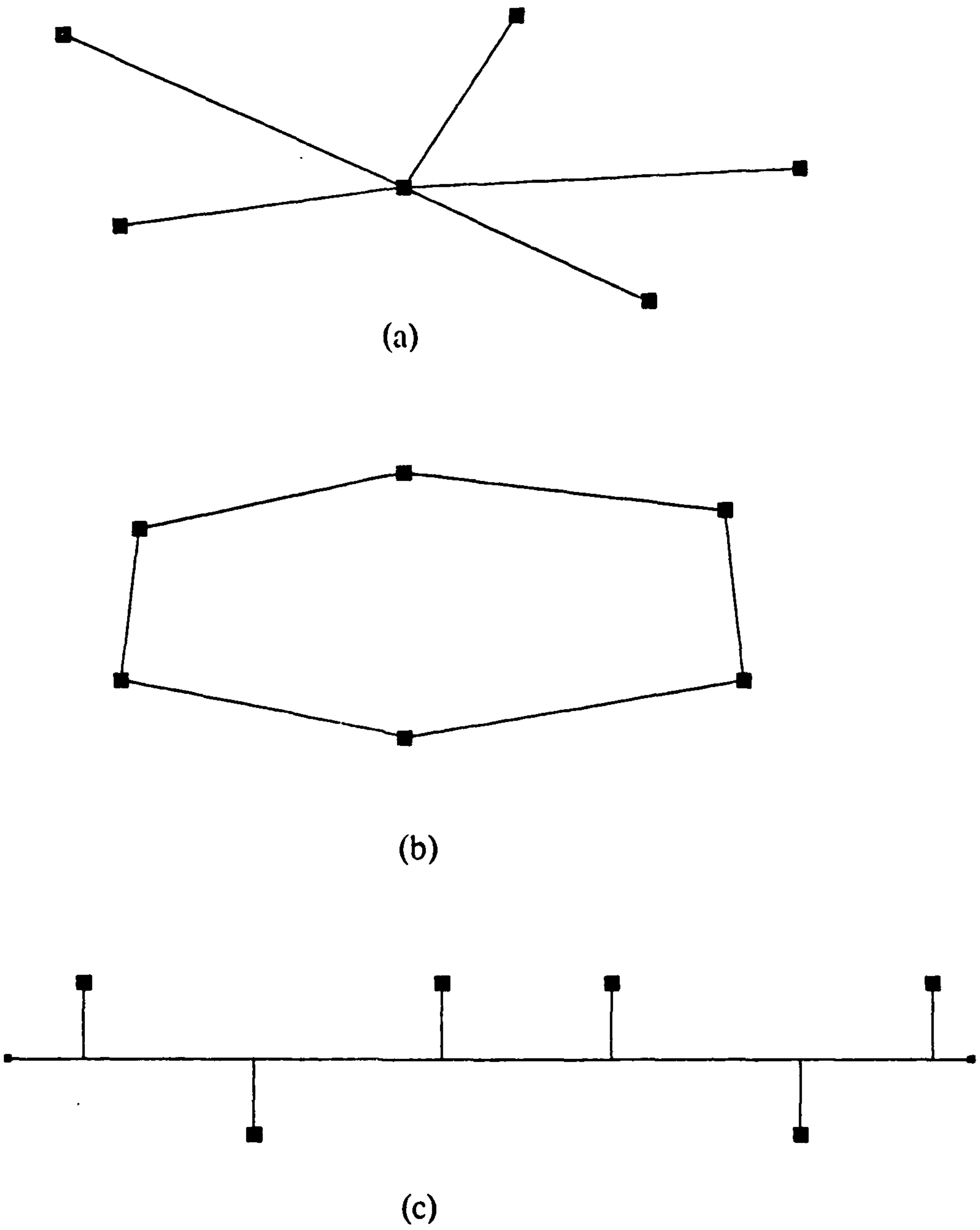


Figure 1-1: Local Area Network Topologies.
(a) Star. (b) Ring. (c) Bus.

achieved by requiring that every packet circulate once around the ring before being removed. The design of protocols for orderly access is aided by the circular point-to-point topology. However, the failure of a single station can disrupt the network unless measures such as redundancy or the use of bypass switches are adopted. Bus networks achieved early prominence owing to reliability and ease of implementation. The bus is passive and the network is unaffected by most failures of individual stations. Stations can be added to and removed from the network during normal operation. In this work, we restrict our attention to bus networks. We note that from the performance point of view, there are commonalities between some ring and bus access methods. Thus, some of our results are indicative of the behaviour of certain ring networks.

Broadcast local area networks had their beginnings in 1970 in a packet-radio network at the University of Hawaii, the *ALOHA* network [Abramson 70]. In ALOHA, remote stations broadcast packets over a common frequency band to a central station. Collisions between multiple packets require retransmissions for reliable data transfer. These are generated by higher level protocols using acknowledgements and timeouts. A more efficient scheme, *carrier sense multiple access (CSMA)*, was proposed by Kleinrock and Tobagi [Kleinrock & Tobagi 75]. In CSMA, stations sense activity on the channel and transmit only when the channel is idle, thus reducing the probability of collision and leading to improved utilization.

With the addition of *collision detection*, the CSMA protocol was implemented in a coaxial cable network, the *Ethernet*, at the Xerox Palo Alto Research Center [Metcalfe & Boggs 76, Crane & Taft 80]. In this variant of the CSMA protocol, *carrier sense multiple access with collision detection (CSMA/CD)*, stations monitor the channel for collisions while transmitting. In the event of a collision, transmission is aborted thus reducing the wastage of bandwidth. The Ethernet was used successfully by a large community of users at a number of interconnected sites [Shoch & Hupp 80]. This led to the introduction of a commercial version of the Ethernet operating at 10 Mb/s [Ethernet 80] and, recently, to the adoption of CSMA/CD as an standard, IEEE 802.3, for local interconnection [IEEE 85a]. The IEEE 802.3 standard and the 10 Mb/s Ethernet are very similar implementations and hereafter we use the term Ethernet to refer to both.

In recent years, large numbers of local area networks, in particular, Ethernets, have been installed. The availability of these networks has made possible experimental performance evaluation both under normal traffic conditions and under artificially generated traffic loads. Measurements on networks with about 100 stations indicated that the average traffic under normal use was less than 100 kb/s [Shoch & Hupp 80, Kume 85]. Even during the busiest 1 second, the traffic was less than 1 Mb/s, well below the typical bandwidths of 10 Mb/s currently in use in local area networks. With the proliferation of networks, however, installations have much larger numbers of stations and new traffic types may have to be accommodated. These factors can cause traffic levels to rise to near the network capacity. Hence accurate performance assessment is imperative.

Taxonomy

In an attempt to impose some order on the burgeoning proposals for local interconnection, various taxonomies have been proposed. Early taxonomies were based primarily on topology and included both local area and store-and-forward networks [Anderson & Jensen 75, Shoch 79]. More recently, it was observed that broadcast networks with differing topologies may actually be very similar in behaviour and could be divided into 4 classes based on the access scheme [Tobagi 80]. In *fixed assignment* schemes, each station transmits over a shared channel in a pre-determined time-slot (TDMA) or frequency-band (FDMA). In *random access* schemes, stations share a channel without any pre-determined assignment. Stations may operate independently (e.g. ALOHA), or may make decisions on when to transmit based on varying amounts of state information from the channel (e.g. CSMA, P-CSMA). The third class is *demand assignment multiple access, or DAMA,* schemes in which bandwidth is allocated in an orderly fashion on demand. This class is subdivided depending on whether control is distributed or centralized. The fourth class covers adaptive strategies in which the protocol varies with load, and hybrid schemes.

Recently, the DAMA class has been further sub-divided into 3 sub-classes based on the nature of the access protocol [Fine & Tobagi 84]. In one sub-class, stations utilize a separate channel or periodic time slots for reservations of the main channel, thus achieving orderly access. In the second sub-class, stations delay transmission by differing times to allow others to gain control of the channel. The delay may be based on station location or other factors and, again, enables orderly access to be achieved. In the last sub-class, stations attempt to transmit at some time after they are ready, monitoring the channel during transmission. In case of conflicts, all stations but one defer transmission. These attempt-and-defer schemes can achieve high performance even at high bandwidths where the performance of the other schemes is poor.

Digitized-Voice Networks

The past decade has seen a trend towards the digitization of voice telephony motivated by the improved noise immunity of digital signals compared to analog signals and the declining cost of digital circuitry [Gold 77]. This has sparked interest in the feasibility of the sharing of communication networks by voice and computer traffic. Such integration can yield economies from the sharing of physical resources, and can ease the use of computer resources such as file servers for voice applications. Further, this can facilitate the integration of voice into traditional data applications such as text editors and electronic mail systems for enhanced functionality [Shoch 80]. Assuming 64 kb/s encoding, a 10 Mb/s network could accommodate no more than 10/0.064 = 156 simultaneously active voice terminals in the absence of data traffic and overhead. Thus, the situation of low utilization of available bandwidth on existing local area network noted above will not hold in the voice/data context. Owing to the differing characteristics and requirements of voice and data traffic, performance evaluations of networks with data traffic are inadequate for voice/data traffic. A need exists for a systematic evaluation of a range of networks with voice/data traffic in order to understand the design trade-offs involved. Alternative approaches being explored for integrated voice/data transmission include (a) providing data access on telephone networks, (b) transmission of packetized voice traffic over computer networks, and (c) hybrid schemes.

A number of papers have addressed issues of voice/data traffic on point-to-point links and centralized switches [Arthurs & Stuck 79, Coviello & Vena 75, Fisher & Harris 76], and metropolitan- and wide-area networks [Maxemchuk & Netravali 85, Weinstein & Forgie 83]. Others have dealt with economic aspects of voice/data networks [Gitman & Frank 78] and general performance issues of voice traffic [Bially *et. al.* 80, Goel & Amer

83, Gruber 81, Gruber & Le 83, Gruber & Strawczynski 85]. Further treatment of the characteristics, requirement and performance measures of voice/data traffic is found in Chapter 2. Prior work on the evaluation of local area networks with voice/data traffic is covered in the next section.

1.2. Prior Work

In this section we review relevant prior work. First, we review studies dealing with data traffic on CSMA/CD. These include analytic modeling, simulation and measurement efforts. Next, we review studies of the performance of broadcast bus local area networks with voice/data traffic, dealing first with contention-based schemes such as CSMA/CD and then with contention-free schemes.

CSMA/CD Performance

Several analytic studies of CSMA/CD performance with typical data traffic have appeared.[1] Metcalfe & Boggs presented a simple formula for estimating the maximum throughput of an Ethernet network [Metcalfe & Boggs 76]. Almes & Lazowska used simulation to characterize the performance of a 3 Mb/s Ethernet and to study the effects of variations in the retransmission algorithm [Almes & Lazowska 79]. Lam used a single-server queueing model to obtain delay-throughput characteristics of CSMA/CD. The technique of embedded Markov chains used to model the CSMA protocol [Kleinrock & Tobagi 75, Tobagi & Kleinrock 77] was extended to CSMA/CD [Tobagi & Hunt 80, Shacham & Hunt 82]. A later study dealt with the effects of carrier detection time in finite population CSMA/CD [Coyle & Liu 83]. Recently, an approximation technique was used to study the performance of an infinite population of stations uniformly distributed on a linear bus [Sohraby *et. al.* 84]. Another approximation technique, equilibrium point analysis, was used to model CSMA/CD with multiple buffers (Chapter 10 in [Tasaka 86]). These studies showed that the CSMA/CD protocol achieves high throughput when the ratio of the propagation delay to the packet transmission time, *a*, is small, less than about 0.1. For larger values of *a*, however, throughput drops significantly,

[1] Some of these are described in greater detail in Section 4.3.1.

e.g., to 10% with $a = 1$. This occurs because a contention overhead on the order of the propagation delay is incurred for each packet while stations learn of each other's transmission attempts.

Few studies of the performance characteristics of actual networks have been reported. Measurements on a 3 Mb/s experimental Ethernet with artificially-generated data traffic with fixed packet lengths showed that high throughput was achieved with packet lengths of 64 bytes of greater. Throughput dropped with shorter packets [Shoch & Hupp 80]. In 1981-82 we extended these measurements to include delay characteristics and a bandwidth of 10 Mb/s [Gonsalves 85]. At 10 Mb/s, high throughput was achieved with packet lengths greater than 500 bytes, but throughput was found to drop to 25% with short packets of 64 bytes. Delay was found to be little greater than the packet transmission time for most of the packets. A few packets, however, suffered delays up to 2 orders of magnitude greater. Our results are discussed further in Section 4.2. Toense described limited measurements on a 1 Mb/s CSMA/CD network with 6 stations generating data traffic [Toense 83], reporting high throughputs owing to the low value of a.

The work described hitherto has provided much knowledge about the behaviour of the CSMA/CD protocol under various conditions. With regard to the Ethernet, the differences between the analytic models and the implementation limit the applicability of these models to the prediction of Ethernet performance. This has been shown by the measurement studies cited above and is described in Section 4.3. The principal difficulty in the analysis of the Ethernet is the nature of the back-off algorithm used to resolve collisions between several packets. Other differences between the implementation and models include the location and number of stations. While some of the analytic studies address some of these issues, no one covers all. Thus, we are led to the use of simulation to further study the behaviour of the Ethernet protocol, particularly at the limits of good performance.

Voice/Data Traffic

Here we review prior work on the behaviour of broadcast bus networks with voice/data traffic. First, we cover random-access schemes and then DAMA schemes.

Nutt & Bayer simulated a 10 Mb/s Ethernet with integrated voice/data traffic, examining the effect of minor variations in the retransmission algorithm [Nutt & Bayer 82]. Tobagi & Gonzalez-Cawley described a simulation study of voice traffic with various encoding rates on 1 and 10 Mb/s CSMA/CD networks [Tobagi & Gonzalez-Cawley 82]. We measured the performance of a 3 Mb/s Ethernet with emulated voice traffic (summarized in Section 4.5.1) [Gonsalves 83]. Musser *et. al.* compared the performance of CSMA/CD and GBRAM [Liu *et. al.* 81], a prioritized form of CSMA, at 1 and 10 Mb/s with only voice traffic [Musser, *et. al.* 83]. DeTreville compared performance of the Ethernet and Token Bus [IEEE 85b] at 10 Mb/s [DeTreville 84]. These studies demonstrated the potential of CSMA/CD for integrated voice/data traffic under low to moderate traffic conditions. The behaviour under heavy traffic was not well examined.

Several proposals have been made for prioritized variants of CSMA and some have been evaluated with voice/data traffic, with voice assigned a higher priority than data [Chlamtac & Eisinger 83, Chlamtac & Eisinger 85, Iida *et. al.* 80, Johnson & O'Leary 81, Maxemchuk 82, Tobagi 80, Tobagi 82]. Most of these schemes retain the contention mode of access, merely restricting the classes of stations that may contend during certain periods to achieve priority. This works well when the high priority class forms a small fraction of the total offered load. In the case of voice/data traffic, voice is expected to comprise the bulk of traffic. Consider the situation when 95% of the traffic is voice and 5% is data. Eliminating the 5% of data traffic via a priority mechanism will not increase the total throughput if the remaining voice traffic continues to use contention access. In particular, at high bandwidths and/or with short packets, when the efficiency of contention access is poor, such a priority mechanism will not help much with the assumed traffic mix. Two of these priority schemes do not suffer from this drawback. Chlamtac & Eisinger propose allocation of alternate frames for voice and data [Chlamtac & Eisinger 85]. Within the voice frame, stations transmit in pre-allocated time-slots. Data traffic uses

CSMA within the data frame. Several issues of synchronization and control are not addressed. Maxemchuk presents an elegant scheme in which voice stations operate in TDMA fashion while data stations contend for the remaining bandwidth [Maxemchuk 82]. Once a voice station obtains access, it is guaranteed access at periodic intervals. Thus the scheme operates efficiently with fixed-rate voice encoding and a prototype has been implemented [DeTreville & Sincoskie 83]. The scheme limits the length of data packets, thus leading to inefficiency in the case of bulk data transfers and is of limited utility in the case of variable-rate encoding or if silence suppression (Section 2.2.1.1) is used to reduce voice bandwidth requirements. The scheme also requires a long packet preamble and hence efficiency drops at high bandwidths.

In addition to the studies listed above, some studies have dealt with voice/data traffic on DAMA networks. Limb & Flamm present a simulation of a Fasnet [Limb & Flores 82] with two 10 Mb/s unidirectional broadcast busses [Limb & Flamm 83]. The traffic is a mix of 64 Kb/s voice channels, with silence suppression, and data packets. In the round-robin Expressnet scheme, the integration of voice and data traffic is facilitated by the use of alternating rounds for the two traffic types [Fratta *et. al.* 81, Tobagi *et. al.* 83]. Fine & Tobagi obtained a simple analytic formula for performance of the Expressnet with fixed-rate voice traffic and a fixed amount of data [Fine 85, Fine & Tobagi 85]. They analysed the case with silence suppression but were not able to obtain numerical results due to exponential growth of the state space. Hence, for this case they used simulation to obtain results for bandwidths of 1, 10 and 100 Mb/s and voice delay constraints of 1, 10 and 100 ms with 64 Kb/s voice sources. It was found that increasing the delay constraint from 1 to 10 ms yields a substantial increase in the voice capcity, i.e., the number of voice sources that can be handled with acceptable quality. A further increase to 100 ms yields a relatively small increase in the voice capacity. The use of silence suppression was found to increase the voice capacity by a factor approximately equal to the ratio of mean talkspurt length to silence duration. The principal limitation of this work is that the heavy traffic assumption is made for data, with each data round being of a fixed length. Thus, the effects of variation in data traffic on voice performance are not studied. Likewise, the performance characterization of data traffic and the effects of voice on it are incomplete.

In this survey of studies of voice/data traffic on local area networks, several points emerge. Firstly, with few exceptions, each study focuses on a single network. Secondly, the degree of detail varies widely, especially between the analytic and simulation studies but also between the different simulation studies. Further, the assumptions regarding traffic characteristics and performance requirements differ. For example, most studies assume a single value for maximum allowable voice delay though this is a subjective parameter and may have a much wider range depending on the application (Section 2.2.1.2). The value chosen ranges from 1 ms to 200 ms in the various studies. Fine & Tobagi consider a range of 1-100 ms for the maximum voice delay for the Expressnet [Fine & Tobagi 85]. Most of the studies, with the exception of Fine & Tobagi, do not consider the use of silence suppression though this has the potential for doubling the number of voice stations that can be accommodated. While much valuable work has been done, the understanding of voice/data networks that emerges lacks in detail and is not comprehensive. In a recent survey of multi-access protocols, Kurose *et. al.* were able to make quantitative comparisons between some protocols with data traffic [Kurose *et. al.* 84]. With voice traffic, however, they were able only to make some general qualitative statements based on results from the literature.

1.3. Contributions

This work addresses two related aspects of local area network performance. The first is the performance of the Ethernet protocol, CSMA/CD, under a wide range of conditions, particularly under high loads. We present measurements on actual Ethernet networks with artificially-generated data and voice traffic (Sections 4.2 and 4.5.1). These demonstrate the potentials of the protocol and its limitations. The protocol is shown to perform well with both data traffic and with emulated voice traffic, but at higher bandwidths and/or under tight delay constraints, performance is poorer. The measurements also show discrepancies between the predictions of prior studies [Metcalfe & Boggs 76, Lam 80, Tobagi & Hunt 80] of the CSMA/CD protocol and the performance of the Ethernet implementation, with the measured performance usually being poorer than the predictions, especially at large *a*.

Due to the limitations of measurement, we resort to simulation, validated with our measurements, to extend the study of the Ethernet protocol as described below. We show that the physical distribution of stations on the network affects individual station performance. In symmetric configurations, stations near the centre of the network obtain a higher than proportionate share of the bandwidth than stations near the ends. In asymmetric configurations, isolated stations are adversely affected (Section 4.4). Next, we study the effects of parameters such as the number of buffers per station and the retransmission algorithm, especially with large numbers of stations (Section 4.3). We show that a simple modification to the retransmission algorithm enables high throughput, close to that predicted by prior analysis, to be achieved even with large numbers of stations. Finally, this study identifies the regions of applicability of analytical models of CSMA/CD for the prediction of Ethernet performance. The simple model of Metcalfe & Boggs [Metcalfe & Boggs 76] is found to be accurate for $a < 0.1$ while the analyses of Lam and Tobagi & Hunt [Lam 80, Tobagi & Hunt 80] are accurate for $a < 0.1$ (Section 4.3.2).

The second focus of this research is the performance of broadcast bus local area networks for integrated voice/data traffic. We propose a new scheme for packetization of voice samples for transmission on broadcast bus networks. By the use of variable length packets, this scheme provides high efficiency at high loads while maintaining low delays at low loads. While several studies of voice/data integration on local area networks have appeared, the differing assumptions made and performance measures used render comparisons with each other difficult. To overcome this, we formulate a network-independent framework for evaluation of such networks in an integrated voice/data context, identifying ranges of interest of various parameters. Owing to the inadequacies of analytic techniques for integrated voice/data networks, we have developed a parametric simulator for such networks and traffic (Chapter 3).

We present a systematic evaluation of representative broadcast bus networks with voice traffic integrated via our variable-length scheme with data traffic. The networks chosen span the range from proven random access schemes to experimental DAMA schemes that operate efficiently at high bandwidths. Specific networks considered are the

Ethernet (IEEE 802.3 standard) and two round-robin schemes, the Token Bus (IEEE 802.4 standard) and Expressnet. We show that deterministic schemes have better performance than random access almost always. Comparison of the performance of the two round-robin schemes highlights the importance of low scheduling overhead, especially at high bandwidths. The trade-offs between two priority mechanisms for round-robin schemes are identified. Numerical results are obtained over wide ranges of key network parameters. Thus, interpolation may be used to obtain approximate performance over a large volume of the design space.

1.4. Overview

In Chapter 2, we discuss the characteristics and requirements of voice and data traffic. This results in the formulation of a consistent set of parameters and performance measures for the evaluation of voice/data networks. In Section 2.2.1.3 we describe our proposed voice packetization protocol. Next, the evaluation methods are discussed in Chapter 3 with particular reference to the networks and traffic types of interest. Chapter 4 contains a characterization of the performance of the Ethernet protocol under diverse conditions. Considering data traffic, we describe a measurement study in Section 4.2. This is extended via simulation to regions in which analytic techniques are inadequate in Section 4.3 and to a consideration of the effects of the physical locations of stations in Section 4.4.

Next, the characteristics of three local area networks with integrated voice/data traffic are described in Section 4.5 (Ethernet), and Chapters 5 and 6 (Token Bus and Expressnet respectively). The emphasis is on aspects specific to particular networks, including the empirical optimization of the parameters of our voice packetization protocol. In Chapter 7 we compare the performance of the three networks with voice/data traffic. Chapter 8 contains a summary of the thesis and conclusions. A summary of notation and details of the simulation methods used, including the program structure and validation, are in the Appendices.

Chapter 2
Framework for Evaluation

To facilitate the comparative study of differing local area networks with integrated voice/data traffic, we formulate a network-independent framework for evaluation. This framework consists of a description of the traffic model and a set of performance metrics. These can be applied to any given network and access protocol as indicated in Figure 2-1. Note that for each network there may be additional metrics of interest, for example, collision counts in CSMA networks. In the remainder of this chapter we discuss the nature of the traffic model and performance metrics and list the values or ranges chosen for various parameters for our evaluation. We also describe the traffic generation processes used and introduce our proposed protocol for voice packetization.

2.1. System Model

The system considered consists of a number of stations interconnected by a network (Figure 2-2). Such a network typically spans a campus or an office building, interconnecting a mix of workstations and voice terminals in individual offices, printing, file and other servers, and time-shared computers. The network may also have one or more gateways to other computer communication networks and to the public telephone network.

The network is characterized by the channel bandwidth, *C*, the topology and the distance spanned. Bandwidths of up to 10 Mb/s are common in operational networks, with experimental networks having bandwidths of 100 Mb/s. Common topologies are linear bus, tree, star and ring topologies. The measure of distance depends on the topology. For a linear bus and a ring, the distance, *d*, is the length of the transmission

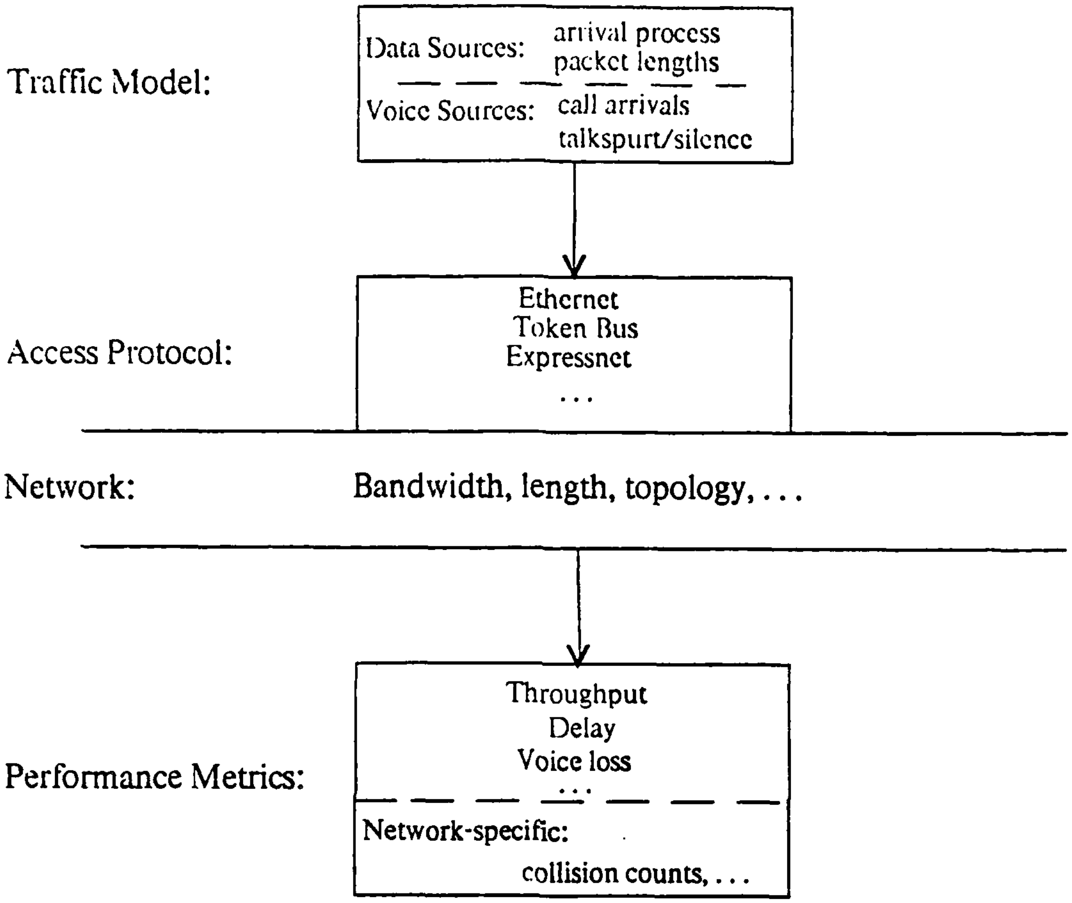


Figure 2-1: A framework for network evaluation

medium and typically ranges between 0.5 - 10 km. For the star and tree topologies a measure of distance is the length of the longest path between any pair of nodes. In the case of unbalanced topologies, the distribution of stations must also be taken into account. In this work we restrict our attention to the linear bus topology and, to a lesser extent, to the star topology.

A voice terminal is considered to be a telephone with digital output. The terminal may be connected to the network via a workstation, sharing the network interface unit of the workstation or it may have its own network interface unit [Shoch 80]. Regardless of the means of connection to the network, the terminal may utilize the power of the

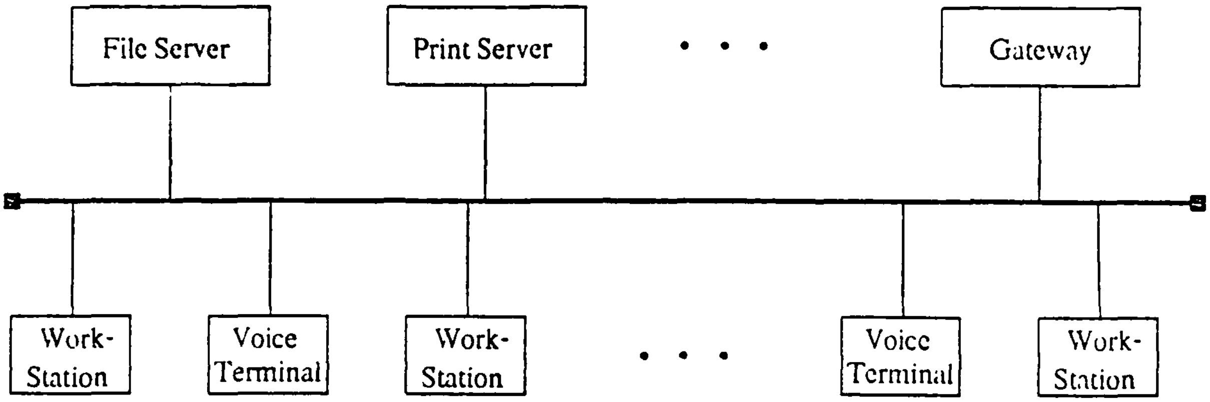


Figure 2-2: A broadcast bus local area network

workstation to provide added functionality, or it may incorporate a dedicated processor and memory [Swinehart, Stewart & Ornstein 83]. In our evaluation, we assume that voice and data stations have individual network interface units and that each station generates only one type of traffic.

2.2. Traffic Model

In this section we describe a traffic model for integrated voice/data networks. For the two types of traffic, voice and data, we describe the characteristics of the traffic and discuss the requirements for acceptable performance. The generation of traffic with specified characteristics is also considered.

2.2.1. Voice Traffic

Voice traffic is assumed to arise from two- or multi-way conversations. Traffic generated by other real-time applications such as the monitoring of remote sensors may have similar characteristics. Note that we do not include here traffic between a person and a voice file-server since, with adequate buffering, such traffic can be modelled as data traffic.

2.2.1.1. Characteristics

The analog speech signal is converted in digital form in the voice station using some coding technique. The coder rate, V, depends on the technique used and may be either fixed or variable. Coding schemes based solèly on the magnitude of the speech signal and its rate of change with time include PCM, DPCM, ADM, ADPCM and typically have constant rates in the range 8-64 Kb/s. These schemes can be used with any analog signal. Lower rate coders are based on the structure of speech and are often referred to as *vocoders.* These have constant or variable data rates as low as 1 Kb/s. Existing and proposed standards for digital telephony specify PCM coding with rates of 32 and 64 Kb/s [Bellamy 82]. We consider only 64 Kb/s.

A voice signal consists of alternating segments of speech and silence. The speech segments, or *talkspurts*, correspond to utterance of a syllable, word, or phrase. The silence segments occur due to pauses between talkspurts. During a two-way voice conversation each speaker alternates between talking and listening, causing additional silence segments. A statistical analysis of 16 conversations showed that each speaker spends about 40-50% of the time talking [Brady 68]. During a conversation, the following states occur for the specified percentages of the total conversation:

One speaker talking:	64-73%
Both speakers talking:	3-7%
Both speakers silent:	33-20%

The ranges above result from varying the threshold used to distinguish silence from speech. Talkspurts of each speaker were found to have mean durations on the order of 1 s while silent intervals were about 50% longer. In both cases, standard deviation was about half the mean.

The talkspurt/silence characteristics of voice may be exploited to achieve increased utilization of channel bandwidth by transmitting the voice signal only during talkspurts. This technique of *silence suppression* is referred to as *time assignment speech interpolation (TASI)* when used to multiplex several analog voice signals on a limited number of circuits [Bullington & Fraser 59]. TASI was developed to maximize the use of trans-Atlantic cables capable of carrying only a few dozen simultaneous circuits.

The operation of a voice terminal can be modelled by a 3-state finite-state machine (Figure 2-3). When no call is in progress, the terminal is in the *inactive* state. During a call, the terminal alternates between two active states, *talk* and *silent*, as described above. At the end of the call, the terminal returns to the inactive state. While talkspurt and silent interval durations are on the order of 1 s, the duration of a call is typically 2 orders of magnitude greater. Thus, the talkspurt transitions and the call on/off transitions may be modelled as separable phenomena. This issue is discussed in some detail by Bially *et. al* [Bially *et. al.* 80]. In our evaluations we assume that all terminals are always active. Based on earlier work [Brady 68], we assume that the times spent in the talk and idle states are uniformly distributed random variables with means 1.2 s and 1.8 s respectively.

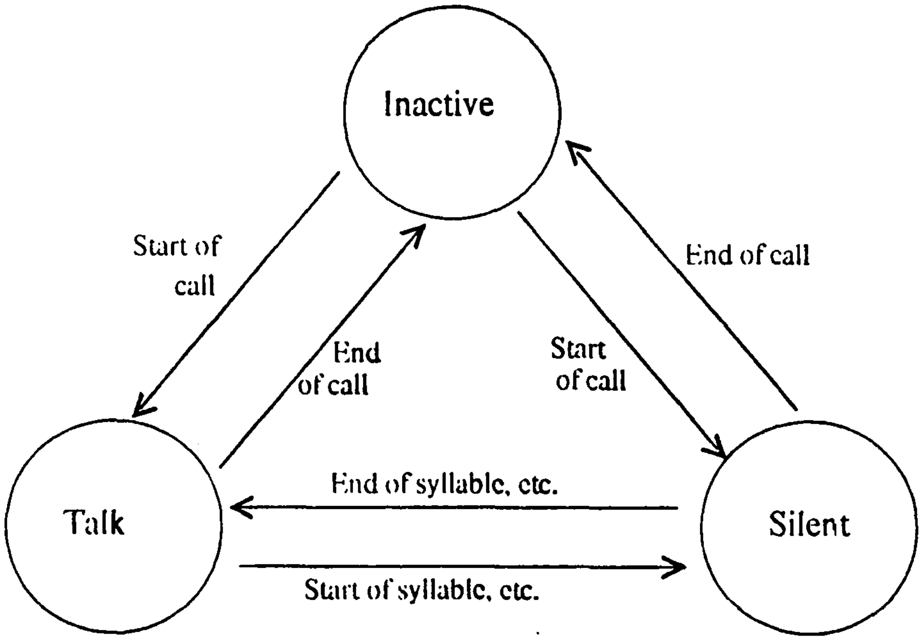


Figure 2-3: State Diagram of a Voice Terminal

2.2.1.2. Requirements

The principal requirement for voice traffic is bounded delay. Voice samples that are not delivered to the receiver within some period, D_{max}, must be discarded. Subject to this constraint, variability of delay is usually acceptable as it can be compensated for by buffering. Typical values for D_{max} depend on the application. For conversations between two local stations, delays of 100-200 ms can be tolerated. For conversations that traverse a public telephone network with a local area network at one or both ends, the delay on each local network should be limited to a smaller value, eg. 10-20 ms, so that the total delay is within acceptable limits. If the delay over the local area network is limited to about 2 ms, the local area network will be indistinguishable from a digital PBX.

Due to congestion, delay in a packet-voice system may exceed D_{max}. In such a case, samples that have suffered excessive delay may be discarded resulting in a loss of some fraction, φ, of voice samples. Owing to the redundancy of speech, low values of loss may not be perceptible to the listener. Studies have shown that losses of up to 1-2% are acceptable [Gruber & Le 83]. The limit depends on the voice coding algorithm used as well as the nature of the loss. If the discarded segments, or clips, are below about 50 ms, the loss appears to the listener as background noise [Campanella 76]. If the discarded segments are larger, syllables or even words may be lost resulting in greater annoyance. For a given loss level, the acceptability of the reconstructed speech decreases as the mean length of the clips increases. Alternatively, to obtain the same quality, greater loss can be tolerated with shorter clips than with longer ones [Gruber & Strawczynski 85]. The annoyance can be reduced by having the receiver replay the latest voice sample received rather than inserting silences [Musser, *et. al.* 83].

2.2.1.3. A Voice Packetization Protocol

In a packet-voice system samples must first be buffered at the transmitter to form a packet which is then transmitted. Thus, delay has two components, the packetization delay and the network delay. The use of shorter packets is desirable to reduce the packetization delay while the use of longer packets is likely to increase utilization of network bandwidth. We propose a variable-length packet protocol that achieves low delay at low network loads and higher efficiency at high loads [Gonsalves 82, Gonsalves 83].[2]

The operation of the protocol is as follows. Each voice station has a first-in first-out (FIFO) packet buffer of some length, P_{max}, in which generated samples are accumulated. When the length of the packet in the buffer reaches a minimum, P_{min}, the station attempts to transmit the packet over the network. While the station is trying to gain control of the network, the packet continues to grow as new samples are generated. When the access attempt is successful, the entire contents of the buffer are transmitted in a single packet. Thus the length of the voice packets varies with time as traffic intensity varies.

While the packet is being transmitted at the channel transmission speed, C bps, it continues to grow at the rate V bps. Thus, in the absence of contention, the length of the shortest packet, P', is greater than P_{min}. During the time taken to transmit P' bytes the packet grows by $(P'/C) \times V$ bytes. Thus, we have:

$$P' = P_{min} + (P'/C) \times V$$

$$i.e., P' = P_{min}/(1 - V/C)$$

The delay is defined to be the time from the generation of the first sample to the successful transmission of the entire packet. The propagation delay over the network is negligible for the cases that we study. The minimum delay is a function of P', $D_{min} = P'/V$

In order to ensure that the delay is bounded, the packet buffer is limited in size to the maximum packet length, P_{max}. If, due to contention, the station cannot transmit the packet before its length reaches P_{max}, the buffer is managed as a FIFO queue, with the oldest sample being discarded when a new one is generated. Thus, we have the maximum delay $D_{max} = P_{max} / \mathrm{V}$.

To summarize, each station accumulates voice samples at V bps until it has a packet of length P_{min}. The optimum value of P_{min} is dependant on the protocol and parameters

[2] A similar approach was reported in a simulation study of the Ethernet and Token Bus for voice transmission [DeTreville 84].

such as P_{max}. It then attempts to transmit, continuing to build the packet until it is successfully transmitted. A maximum, P_{max} is imposed on the packet length to bound the delay to D_{max}. However, this can cause loss of samples due to network congestion. At low loads, packets are short leading to delays little greater than D_{min}. At high loads, packets are long, improving utilization due to amortization of the protocol overhead per packet over a large number of voice samples. In addition, in protocols such as CSMA/CD, utilization is an increasing function of packet length.

2.2.2. Data Traffic

Data traffic is assumed to arise from computer communication applications. These include *interactive* applications such as remote logins and transaction processing. Non-interactive or *bulk* traffic arises from applications such as file transfers and electronic mail.

2.2.2.1. Characteristics

Data traffic is typically *bursty* in nature, i.e., a station alternates between periods of high network activity separated by relatively long periods during which it generates few packets. The traffic may be characterized by the packet arrival process and the packet length distribution.

The packet length distribution is a function of the application environment and the protocols used and may be expected to vary widely. In an experimental study of normal traffic on an operational Ethernet interconnecting over 100 workstations and several servers in the Computer Systems Laboratory at the Xerox Palo Alto Research Center, the distribution of packet lengths was found to be bimodal [Shoch & Hupp 80]. The shorter packets, of length about 32 bytes, consisted almost entirely of protocol overhead with a few bytes of data. Such packets are generated by interactive applications and protocol control functions. The longer packets, of length between 512 and 576 bytes, resulted from bulk data transfer applications with the packet length being the maximum allowed by the protocol. The ratio of the number of short packets to the number of long packets was such that the short packets comprise about 15-30% of the total data bytes. The numbers quoted here are specific to a particular application environment and protocol. However, the bimodal distribution and the abundance of short relative to the number of long packets is expected to be more general. We denote the packet length by P, and the mean packet transmission time by $T_p = P/C$.

The packet arrival process depends not only on the application but also on the protocol implementation and *network interface unit* in the station. We assume that the network interface unit has a *transmission buffer* in which the packet is stored while the transmission attempt is being made. There may be additional buffers to queue packets. The total number of buffers in a network interface unit is denoted by B.

Two modes of behaviour may be identified, *non-feedback* and *feedback*. In the former, packet arrivals are independent of one another. This mode is approximated in a multi-tasking system with several independent tasks may be generating packets concurrently. If one task is blocked because it is waiting for a free packet buffer, other tasks may run and also generate packets. Thus, the arrival of packets to the network interface unit is less dependent on the current state of the buffers. Note that the non-feedback mode may also occur in applications where packets are broadcast periodically with some information such as the time or the status of some instrument. In these cases, packets are simply discarded when the buffer is full. In contrast, in the feedback mode, the generation of a new packet starts only when the previous one has been transmitted.[3] Thus, if the buffers are full, the station must wait until a packet has been transmitted before it can generate another packet. This mode of operation is likely to occur in single-tasking systems and with interactive applications.

A simple model of the non-feedback mode is shown in Figure 2-4. The operation of the access protocol in each station is modelled by a network server. This server includes the transmission buffer of each network interface unit. The queue of this server is thus physically distributed. Its service rate and discipline are functions of the protocol. The

[3] The feedback mode usually arises when there is only one packet buffer. For the sake of completeness, we generalize our definition to the multiple-buffer case.

service rate is also, in general, a function of the number of packets in the queue. In some protocols, all packets eventually receive service and contribute to the throughput, η. Packets may also be discarded after some time in the network server due to congestion and do not contribute to η. In this case, the network server, may be represented by two stages, Net Server A and Net Server B. The former corresponds to the service until access is obtained or until the packet is discarded, and the latter corresponds to successful transmission of the packet. Packets arrive at the ith network interface unit at the rate $1/\theta_i$. If the buffers are full, packets are discarded. Note that while the ith network interface unit has B_i buffers, only B_i - 1 are shown in the model of the network interface unit, the remaining one, the transmission buffer, being in the central network server.

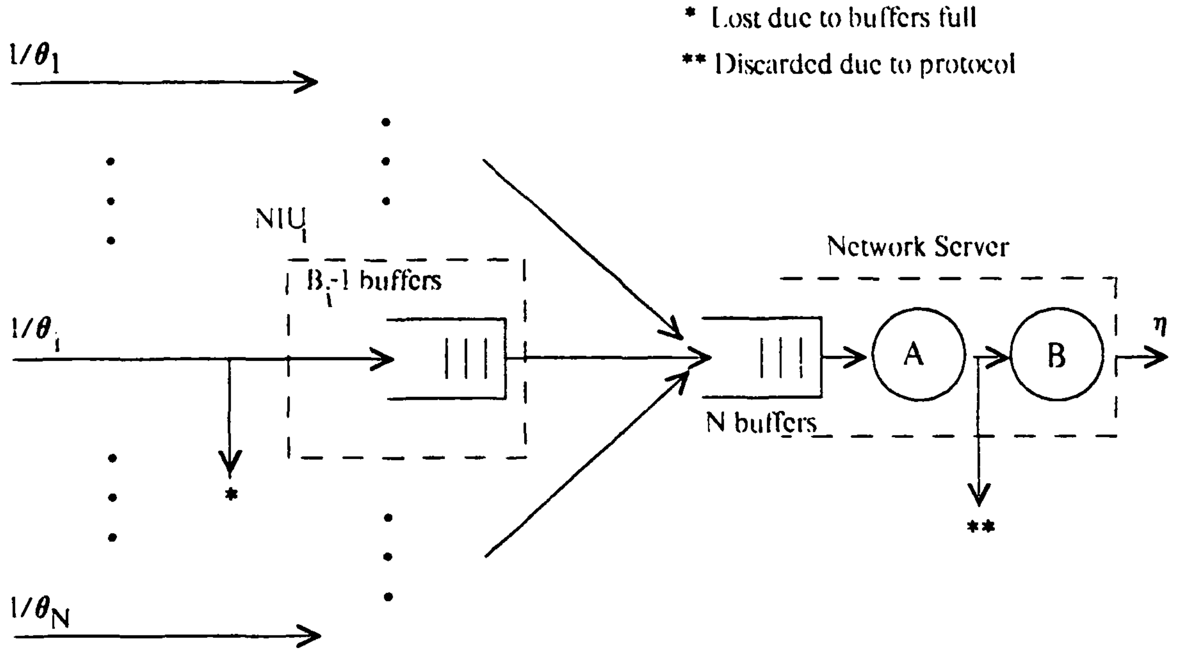


Figure 2-4: Packet Arrival Process: Non-Feedback mode.
B_i buffers in station i, $1 \le i \le N$.

We define *offered load*, G_i, of a station i to be the rate at which traffic enters the network from NIU_i if the network server had infinite capacity.[4] There is no blocking and throughput is equal to the offered load. The offered load of all N stations, is defined to be:

$$G = \Sigma_i G_i \tag{2.1}$$

G is thus the mean service rate of the Norton equivalent server of the model excluding the network server [Chandy *et. al.* 75]. G_i is clearly equal to $1/\theta_i$ packets/second. For convenience, we represent G as a percentage of channel capacity, C. Each packet contributes on the average P_d bytes of useful data. The transmission time of this is T_{p_d} = P_d/C. Thus, we have:

$$G_i = T_{p_d}/\theta_i \times 100\% \tag{2.2}$$

Packet delay, D, is defined to be the time from when the packet first enters the network interface unit to the time at which it leaves the network server. There are two components to delay: the time spent queued in the B_i-1 buffers in the network interface unit and the time spent queueing for and receiving service at the network. The latter is often referred to as congestion delay or service time, while the former is classical queueing delay.

The feedback mode is modelled with a closed queueing network (Figure 2-5). Station i has B_i *jobs* in class i. After a class i job receives service at the network, it enters *station server i*. This server corresponds to the various levels of software which generate packets and has service rate $1/\theta_i$. During periods of network congestion, all the jobs of a station may be in the NIU and network server, and the station server is idle.

Offered load, throughput and delay are defined as in the non-feedback case.

[4] This usage of G differs from the conventional usage to denote the channel offered traffic in infinite population analyses [Kleinrock 76]. In the latter, G denotes the rate at which packets, new or previously collided, are scheduled for transmission on the channel.

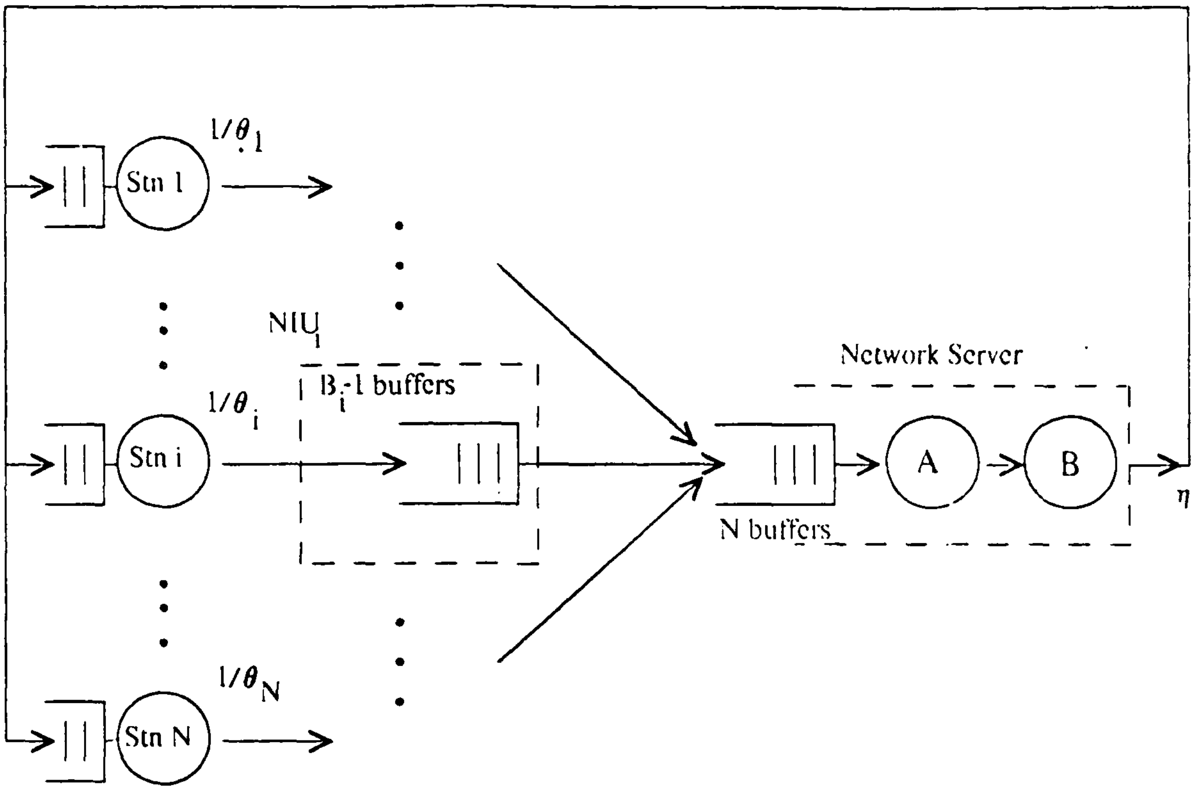


Figure 2-5: Packet Arrival Process: Feedback mode.
B buffers and *i* jobs in station *i*, $1 \le i \le N$.

2.2.2.2. Requirements

Data traffic requirements vary with the application. Delays of several seconds are often acceptable in cases such as bulk file transfers and electronic mail. Some interactive traffic also can tolerate such delays. When the network is used to login to a remote system, the delay requirement is dependant on whether character echoing is performed remotely or locally. In the former, average packet delay should not exceed about 100 ms, the time taken for a fast typist to type one character. If echoing is done locally, higher delays can be accepted. In some systems, characters are echoed and buffered locally, with transmission to the remote system occurring only when one or more lines have been accumulated. Here, the delay requirements are much less stringent. Variability of delay is again usually acceptable except in cases of remote echoing.

Reliability of transfer is important only from a performance point of view for most higher level protocols as these protocols implement measures to ensure reliable data transfer [Saltzer, Reed & Clark 84]. However, in protocols such as in the V operating system, high reliability is crucial to performance as the V communication protocols are based on the assumption of a highly reliable, high speed local area network [Cheriton 83]. For bulk data transfers, some minimum average throughput is desirable in order to limit end-to-end delay.

2.2.2.3. Generation

Traffic patterns on local area networks during normal usage have not been thoroughly characterized and may be expected to vary widely between installations. Thus the choice of traffic parameter values in an evaluation is of secondary importance compared to the consistent use of the same set of parameters for all networks and experiments. In most of the simulation experiments, the aggregate offered data traffic, G_d, is assumed to be some multiple of a *standard data load*. The standard data load is here defined such that G_d = 5% of 10 Mb/s. Thus, 2 standard loads correspond to G_d = 10%, and so on. 10 Mb/s was chosen as the baseline as this is the lowest channel capacity likely to be of interest given data rates of 32 - 64 Kb/s per voice station. For higher capacity networks we use multiples of the standard to achieve the desired data traffic loading.

Based on the measurements cited in Section 2.2.2.1, we use the bimodal distribution for packet length with values of 50 and 1000 bytes for *interactive* and *bulk* packets respectively. These values do not correspond to any particular protocol but are representative of several [Boggs *et. al.* 80, Ethernet 80]. For our standard, we assume that 20% of the data bytes are carried in 50-byte packets, with the remaining 80% being in 1000-byte packets.

In order to generate 100 Kb/s (20% of 5% of 10 Mb/s) of interactive traffic, we can use several sets of values for the number of stations, N_i, and their offered load G_i. Likewise, to generate 400 Kb/s of bulk traffic we can chose sets of values for N_b and G_b. It seems reasonable to assume that typically N_i is larger than N_b. We use N_i:N_b :: 2:1. The larger the value chosen for N_i, the larger will be the corresponding value of θ (from Equations (2.1) and (2.2)). In order to obtain statistically significant results, θ must be much smaller than the simulation run time. Hence we chose the smallest possible values for N_i and N_b, i.e. 2 and 1 respectively. These yield values of G of about 20 and 10 ms. The stations operate in the non-feedback mode. To achieve a load of n standard loads, we use $2n$ interactive stations and n bulk stations.

2.3. Performance measures

For connection-based traffic such as voice, three phases can be identified: the *access phase*, the *information transfer phase*, and the *disengagement phase* [Gruber & Le 83]. During the access phase an attempt is made to setup a circuit, physical or virtual, between the two ends. If the connection is successfully established, information transfer can take place. Thereafter, the connection must be broken. Several performance measures of interest are associated with each of the phases. For example, during the access phase, measures of interest include the access time, the probability of incorrect access, and the probability of access denial. A treatment of these issues is given by Gruber & Le, cited above. In this work, we are concerned primarily with performance during the information transfer phase. Note that in the case of datagram-based data traffic this is the only phase.

2.3.1. System

The primary performance measure for the system is utilization of the channel bandwidth under various traffic conditions. It is desirable to maximize utilization as system cost increases with bandwidth because higher speed circuitry is more expensive than lower speed circuitry. The cost of the cable is less strongly dependant on bandwidth.

A given network usually has additional metrics of interest. In the case of the Ethernet, the distribution of the number of collisions suffered by a packet and the fraction of packets discarded due to excessive collisions yield insights into the performance of the back-off algorithm. In prioritized protocols, such as the Token Bus and Expressnet, the efficiency of the priority mechanism is of importance. These issues are discussed further in Chapters 4, 5 and 6.

2.3.2. Voice

The primary voice performance measure is the fraction of samples lost, φ. Given constraints on the maximum acceptable loss, φ, and maximum acceptable delay, D_{max}, we denote by $N_{v_{max}}^{(\varphi,D_{max})}$ the maximum number of voice stations that can be simultaneously accommodated. In the interests of clarity, we drop one or both of the superscripts when the value is clear from the context. $N_{v_{max}}^{(\varphi,D_{max})}$ is also a function of other parameters such as G_d. To the user, delay is unimportant as long as it is below the threshold of acceptability. Once this threshold is exceeded, delay becomes important. To the system designer, delay is important even below the threshold. Depending on the way in which delay increases with increasing system load, the designer may be able to mitigate the effects by suitable buffering and playback schemes at the receiver. Such schemes may also be necessary to compensate for large variance in delay. We restrict our attention to the behaviour of the transmitter.

2.3.3. Data

Data traffic performance measures of importance are average throughput and delay. Variance of delay is of importance for some applications. In such cases, delay histograms and percentiles can be useful. In computing η_d we usually lump both interactive and bulk data traffic together. For delay we distinguish between the two types since it is not necessarily meaningful to average over either the number of packets or the number of bytes.

2.4. Summary of Parameter Values

This section contains a summary of the parameters used to represent a system in our evaluations. For each parameter, a range or a single value is specified. These are based on typical cases, fundamental limitations, and existing or proposed standards as discussed in the preceding Sections.

2.4.1. System Parameters

The system parameters used are summarized in Table 2-1.

Network Parameters:	
Length, d	1, 5 km
Bandwidth, C	10, 100 Mb/s
Signal propagation delay	0.005 μs/m
Topology	Linear bus
Access Protocol	CSMA/CD, Token Bus, Expressnet
Station Parameters:	
Packet overhead, P_o	10 bytes
Packet preamble, P_p	64 bits
Packet buffers	1
Carrier detection time, t_{cd}	10 bit transmission times, i.e.. 1.0 μs at C = 10 Mb/s 0.1 μs at C = 100 Mb/s
Inter-frame gap, t_{gap} (Ethernet, Token Bus)	100 bit transmission times, i.e., 10.0 μs at C = 10 Mb/s 1.0 μs at C = 100 Mb/s

Table 2-1: System Parameters: Values Used

2.4.2. Voice Traffic Parameters

The parameters used for each voice station are shown in Table 2-2. The value of D_{min} is empirically optimized for each protocol (Chapters 4, 5 and 6).

Voice Traffic Parameters:

Encoding rate, V	64 Kb/s (constant)
Maximum delay, D_{max}	2, 20, 200 ms
Minimum delay, D_{min}	$< D_{max}$ (protocol-dependent)
Silence suppression	yes, no
Mean talkspurt length, t_t	1.2 s (uniformly distributed)
Mean silence length, t_s	1.8 s (uniformly distributed)

Table 2-2: Voice Traffic Parameters: Values Used

2.4.3. Data Traffic Parameters

The parameters used for data traffic are summarized in Table 2-3. The packet length and arrival parameters shown are used to generate 1 *standard load* of G_d = 0.5 Mb/s using 3 stations. Multiples of the standard load are generated by increasing the number of stations appropriately.

Data Traffic Parameters:

G_d	0-50% of network bandwidth, C
Packet length, P	*Interactive*: 50 bytes (constant) *Bulk*: 1000 bytes (constant)
Fraction of bulk traffic	80% of G_d (by volume)
Fraction of interactive traffic	20% of G_d (by volume)
Standard Data Load	0.5 Mb/s
θ	*Interactive*: 7.9 ms (uniform) *Bulk*: 19.2 ms (uniform)

Table 2-3: Data Traffic Parameters: Values Used

2.5. Summary

We have discussed the characteristics, requirements and performance measures typical of local area networks, computer communication traffic and packetized voice telephony. This has led to the formulation of a network-independent framework for evaluation of voice/data networks. Based on criteria such as measurements and standards, we have selected values or ranges for various parameters for use in our evaluations. The variable parameters are channel bandwidth and length, access protocol, the ra.io of data and voice traffic, the maximum allowable voice delay and the use of silence suppression. Important performance measures are system throughput, delay and throughput of data traffic, and loss and the maximum number of voice stations under given constraints.

Chapter 3
Evaluation Methodologies

Performance evaluation methodologies can be divided into three types: analytic modelling, simulation and measurement. In analytic modelling, a mathematical model of the system under study is constructed. This model is then solved to obtain equations for the performance measures of interest.[5] Analytic solutions are sometimes easily obtained and allow study of various design alternatives. This can provide insights into the effects of key parameters. For all but simple cases, simplifying assumptions are necessary for tractability, thus resulting in possible inaccuracies. For complex systems, computationally expensive iterative techniques may be necessary to obtain numerical results.

Simulation can be used to model a system to any desired degree of detail. There is a trade-off between detail on the one hand and on the other, programming effort and computer run time. The validity of the simulator is also an issue. The third technique, measurement on actual systems can yield the most accurate performance assessment. This approach lacks flexibility and may be expensive. The detail possible with simulation and measurement may actually hinder understanding by masking important trends. Thus, design of the model is important.

In the rest of this Chapter, we discuss the advantages and disadvantages of the three techniques for the performance evaluation of local area networks within the context of the framework developed in Chapter 2 and discuss the techniques used in our evaluations. For the reader unfamiliar with the field, several books deal with performance evaluation methodologies with applications to computer systems [Ferrari 78, Kleinrock 75, Kleinrock

[5] We follow customary usage of the term analytic to refer to all cases when numerical results are obtainable by means other than simulation or measurement.

76, Kobayashi 81]. A recent survey by Heidelberger & Lavenberg covers post-1970 developments in the field [Heidelberger & Lavenberg 84].

3.1. Analytic Techniques

We now discuss the applicability of analytic techniques for performance evaluation of local area networks with voice/data traffic and then for the Ethernet network. As was discussed in Section 1.2, several analytic studies have dealt with these topics, particularly the latter, and have provided considerable knowledge in the area. There are, however, limitations to the analytic method.

Several assumptions are necessary in analytic modelling for tractability. One of these is that packet arrivals form a Poisson process. This is often valid for typical data traffic, but does not match the nature of voice traffic well since voice samples are generated at a constant rate. It also does not match well the arrivals of bulk data traffic, especially when the number of stations is small. Thus, many of the useful analytic techniques cannot be applied to voice traffic.

A network with round-robin scheduling carrying only voice traffic is deterministic if silence suppression is not used and hence simple equations for performance can be obtained. If silence suppression with randomly varying talkspurt durations is used, however, the situation is more complex. In an analysis of voice traffic with silence suppression on the Expressnet, the state space was found to grow exponentially and to be impracticably large, on the order of 10^{28}, for even unrealistically small systems [Fine & Tobagi 85].

Analytic models of broadcast local area networks often require that operation be slotted in time to reduce the complexity of the analysis. All stations are assumed to be synchronized and to start transmission only on slot boundaries. In the case of asynchronous protocols such as CSMA and CSMA/CD, this leads to optimistic predictions of performance. This also usually implies that station locations cannot be taken into account since stations on a linear bus observe events at times dependant on the propagation delay between stations. These limitations can be overcome at the expense of increased complexity in the analysis or by the use of approximation techniques. With the former, obtaining numerical results may be difficult, with the latter, accuracy is an issue.

Considering the CSMA/CD protocol, one of the determinants of performance is the algorithm used to determine the delay before a retransmission attempt upon detection of a collision. Analytic models typically assume that the an optimum algorithm is used such that the probability of a retransmission in every slot is constant, i.e., the time interval until the retransmission attempt is geometrically distributed. The Ethernet implementation, diverges from this model in several respects in an attempt to provide good performance at low loads and stability at high loads. Firstly, the policy followed changes with the number of collisions suffered by each packet to adapt to varying loads. Secondly, to limit delay, packets are discarded after some maximum number of collisions (this is necessary in any practical implementation). As we will show, these differences affect performance significantly (Section 4.3.3).

The above shortcomings of analytic modeling with respect to the Ethernet protocol and voice/data traffic can be overcome by the use of simulation and experimental measurement. This serves to obtain accurate and realistic characterizations of performance and to study aspects that cannot be modelled analytically. In the following paragraphs, we discuss some details of the simulation and measurement techniques used in our study.

3.2. Simulation

Validation of a simulator is important both to ensure that the model chosen is a faithful representation of the system being modelled, a concern also in analytic modelling, and to ensure that the program is free of significant errors. The use of sound programming techniques and of careful and extensive testing served to increase our confidence in the correctness of the program. For our Ethernet simulator, we validated the simulation results with our measurements on actual systems. Good correlation was obtained with residual differences being attributable to factors such as variable circuit delays that were only crudely modeled. For the other protocols, for which accurate analytic models are available for certain cases, we validated the simulation with such models. Details are presented in Appendix B.2 along with a description of the structure of the program in Appendix B.1.

In a stochastic simulation, estimation of the accuracy of performance measures is necessary [Kobayashi 81]. The measure of accuracy used is the confidence interval at a specified confidence level. In a broadcast local area network under moderate and heavy loads, i.e., with a large number of stations and/or closely-spaced packet arrivals, the regenerative method for obtaining confidence intervals is impractical. Regeneration points are difficult to detect and may be expected to occur infrequently. We resort to the method of sub-runs for obtaining confidence estimates. For each run, the simulator is run for an *initial transient* period before any data are collected to allow the system to reach steady state. The duration of the transient period is dependant on parameters such as the bandwidth and maximum voice delay and is determined empirically. Values used range from 1 s to 10 s. After the transient period, the simulator is run for *n* consecutive sub-runs of duration *t* s each. Performance measures are obtained separately for each sub-run and these are used to estimate confidence intervals. Simulation run times, *nt*, ranged between 5 and 100 s, depending on system parameters, with *n* ranging up to 10 sub-runs. These times yielded 95% confidence intervals of less than 1% of the mean for aggregate statistics in most cases.

3.3. Measurement

Experiments on a broadcast local area network can be conveniently controlled from a single station on the network. With appropriate software in the stations, it is possible to find idle stations on the network and to load a test program into each from the controller [Shoch & Hupp 80]. In the absence of such software it is necessary to load the test program into each of the participating stations manually. The controller is then used to set parameters describing the traffic pattern to be generated by the test programs. Next, the test programs are started simultaneously by a message broadcast by the controller. They generate traffic and record statistics for the duration of the run. To ensure complete overlap of the data collection periods in all stations, it is necessary to have the test programs run for some period before and after the data collection period. At the end of the run, the statistics are collected from the participating stations by the control program. Empirical tests on the setup used for our Ethernet measurements indicated that there was no significant variation in statistics for run times between 10 and 600 s. For most experiments we use run times between 60 and 120 s.

3.4. Summary

A consideration of the nature of the problem of evaluation of local area networks reveals several difficulties in the application of analytic techniques to voice traffic and to the Ethernet protocol. This view is supported by the preponderance of simulation studies of voice/data traffic in the literature (Section 1.2). Thus, while analytic models are used in some instances, the techniques of choice for our study are measurement and simulation. The former is especially important in the case of the Ethernet protocol which has several aspects not amenable to analytical modeling. It thus serves both to accurately characterize performance and to provide a means of validation of simulation models.

Chapter 4
Ethernet

Recently, the Ethernet has come into widespread use for local interconnection.[6] With increasing usage, it is expected that networks will have to support large numbers of stations and new traffic types. Hence, in this Chapter, we use experimental measurement and simulation to obtain a detailed characterization of the performance of the Ethernet protocol under varied conditions and traffic types. In Section 4.2, we present some results of measurements on a 3 Mb/s experimental Ethernet and a 10 Mb/s Ethernet performed at the Xerox Palo Alto Research Centers in 1981-82. These experiments show the potentials and limitations of the protocol. By the use of measurement, we obtain measures such as delay distributions that provide new insights into the behaviour of the protocol under adverse conditions.

The Ethernet protocol, CSMA/CD, has been the subject of several analytic studies. While the studies have resulted in improved understanding of several aspects of the CSMA/CD scheme, certain aspects of the Ethernet implementation, in particular, the retransmission algorithm, are not easily amenable to analytic modeling. Comparison of the analytical predictions of CSMA/CD performance with our measurements show a correspondence ranging from good to poor depending on parameter values. Given the difficulty of altering parameters in operational networks, we resort to the use of a detailed simulation model to improve our understanding of the protocol in regions in which analysis fails. We show that a simple modification to the retransmission algorithm enables higher throughputs to be achieved with large numbers of stations than the standard

[6] We note that the Ethernet [Ethernet 80] and the IEEE 802.3 standard [IEEE 85a] are very similar. We use the term *Ethernet* to refer to both.

protocol (Section 4.3). Since the throughput of the modified algorithm is close to that predicted by prior analysis using optimum assumptions, we conjecture that the modified algorithm is near-optimal.

The performance of the Ethernet protocol is dependant on the propagation delay between stations. Thus, it is expected that the distribution of stations on the network will influence performance. In Section 4.4, we present a simulation study of the effects of various distributions of stations on a linear bus Ethernet. The distributions include balanced and unbalanced ones.

Finally, an important new traffic type considered for local area networks is packetized voice. By the use of measurement, we show that real-time traffic can be successfully carried on a 3 Mb/s Ethernet. These measurements were made on an experimental Ethernet at the Xerox Palo Alto Research Center in 1981. We then extend this work via simulation to integrated voice/data traffic at higher bandwidths using the evaluation framework of Chapter 2 (Section 4.5).

4.1. The Ethernet Protocol

We now describe the Ethernet architecture and details of two specific implementations, with bandwidths of 3 and 10 Mb/s, for which we report experimental measurements. The description is limited to the features that are relevant to the evaluation. Startup, maintenance and error-handling procedures are not described since these are expected to be invoked infrequently in local area network environments. For further details, the reader is referred to the literature [Metcalfe & Boggs 76, Ethernet 80, IEEE 85a].

4.1.1. The Ethernet Architecture

The access protocol used in the Ethernet is *carrier sense multiple access with collision detection (CSMA/CD).* Carrier sense multiple-access (CSMA) is a distributed scheme proposed to efficiently utilize a broadcast channel [Kleinrock & Tobagi 75]. In CSMA, a host desiring to transmit a packet waits until the channel is idle, i.e., there is no carrier, and then starts transmission. If no other hosts decide to transmit during the time taken for the first bit to propagate to the ends of the channel, the packet is successfully transmitted (assuming that there are no errors due to noise). However, due to the finite propagation delay, some other hosts may also sense the channel to be idle and decide to transmit. Thus, several packets may collide. To ensure reliable transmission, acknowledgements must be generated by higher level protocols. Unacknowledged packets must be retransmitted after some timeout period for reliable data transfer.

In order to improve performance, the Ethernet protocol incorporates collision detection into the basic CSMA scheme. To detect collisions, a transmitting host monitors the channel while transmitting and aborts transmission if there is a collision. It then *jams* the channel for a short period, t_{jam}, to ensure that all other transmitting hosts also abort transmission. Each host then schedules its packet for retransmission after a random interval chosen according to some *retransmission or back-off algorithm*. The randomness is essential to avoid repeated collisions between the same set of hosts. The retransmission algorithm can affect such characteristics as the stability of the network under high loads, delays suffered by packets and the fairness to contending stations. The incorporation of retransmissions in the network interface in the Ethernet enable much faster response to collisions than in CSMA where the retransmission depends on timeouts in higher level software.

4.1.2. A 3 Mb/s Ethernet Implementation

We summarize the relevant details of the 3 Mb/s Ethernet local network used in our experiments. This network has been described in detail earlier [Crane & Taft 80, Metcalfe & Boggs 76]. The network used in our experiments has a channel about 550 metres long with baseband transmission at 2.94 Mb/s. The propagation delay in the interface circuitry is estimated to be about 0.25 μs [Boggs 82]. The end-to-end propagation delay, τ_p, is thus about 3 μs. The retransmission algorithm implemented in the Ethernet hosts (for the most part, Alto minicomputers [Thacker, *et. al* 82]) is an approximation to the binary exponential back-off algorithm. In the binary exponential back-off algorithm, the mean retransmission interval is doubled with each successive collision of the same packet. Thus, there can be an arbitrarily large delay before a packet is transmitted, even if the network is not heavily loaded. To avoid this problem, the Ethernet hosts use a truncated binary exponential back-off algorithm. Each host maintains an estimate of the number of hosts attempting to gain control of the network in its load register. This is initialised to zero when a packet is first scheduled for transmission. On each successive collision the estimated load is doubled by shifting the load register 1 bit left and setting the low-order bit to 1. This estimated load is used to determine the retransmission interval as follows. A random number, χ, is generated by ANDing the contents of the load register with the low-order 8 bits of the processor clock. Thus, χ is approximately uniformly distributed between 0 and 2^n-1, where n is the number of successive collisions, and has a maximum value of 255, which is the maximum number of hosts on the network. The retransmission interval is then chosen to be χ time units. The time unit should be no less than the round-trip end-to-end propagation delay. It is chosen to be 38.08 μs for reasons of convenience. After 16 successive collisions, the attempt to transmit the packet is abandoned and a status of load overflow is returned for the benefit of higher level software.

Thus, the truncated back-off algorithm differs from the binary exponential back-off algorithm in two respects. Firstly, the retransmission interval is limited to 9.7 ms (255×38.08 μs). Secondly, the host makes at most 16 attempts to transmit a packet.

4.1.3. A 10 Mb/s Ethernet Implementation

The 10 Mb/s Ethernet used is similar to the network described in the previous section with a few exceptions. The channel consists of three 500-metre segments connected in series by two repeaters. The end-to-end propagation delay, including delay in the electronics, is estimated to be about 15 μs (see pg. 52 in [Ethernet 80]). The truncated binary exponential back-off algorithm uses a time unit of 51.2 μs. The load estimate is doubled after each of the first 10 successive collisions. Thus the random number, χ, has a maximum value of 1023, yielding a maximum retransmission interval of 53.4 ms. A maximum of 16 transmission attempts are made for each packet.

4.2. Data Traffic: Measured Performance

In this section we present the results of our experiments. First, we describe the experimental set-up and procedures, and the traffic patterns used. Then we discuss the performance of the 3- and 10-Mb/s Ethernets separately. Finally, we make some comparisons between the two sets of experiments.

4.2.1. Experimental Environment

Experiments were setup and run using the procedure described in Section 3.3. We note that in the 3 Mb/s Ethernet experiments, the design of the operating system of the stations, Alto minicomputers [Thacker, *et. al* 82], allowed us to run the entire experiment from a single control station [Shoch & Hupp 80]. This included location of idle stations and loading of the test programs over the network. On the 10 Mb/s Ethernet, however, loading of the test programs was done manually in each station. The duration of each run was 60 - 120 s. Our tests show that there is no significant variation in statistics for run times from 10 to 600 s.

Measurements on the 3 Mb/s Ethernet [Shoch & Hupp 80] and our informal observations of traffic on the 10 Mb/s Ethernet indicate that at night the normal load rarely exceeds a small fraction of 1% of the network capacity. Thus, it is possible to conduct controlled experiments with specific traffic patterns and loads on the networks during the late night hours.

We ignore packets lost due to collisions that the transmitter cannot detect ([Shoch 79], p. 72) and due to noise since these errors have been shown to be very infrequent [Shoch & Hupp 80]. That is, we assume that if a packet is successfully transmitted it is also successfully received. Thus, all the participating stations in an experiment are transmitters of packets. In computing throughput, η, we assume that the entire packet, except for a 6-byte header and checksum[7], is useful data. Thus, our results represent upper bounds on performance since many actual applications will include additional protocol information

[7] We assume 4 bytes of overhead in the case of the 10 Mb/s Ethernet

in each packet. Packets whose transmission is abandoned due to too many collisions are not included in the mean delay computations.

4.2.2. 3 Mb/s Experimental Ethernet

In this section we describe the measured performance characteristics of a 550 m, 3 Mb/s Ethernet. In all the experiments reported, the number of stations, N, was 32. Fixed length packets were used with the inter-arrival times of packets at each station being uniformly distributed random variables. Stations were operated in the feedback mode with a single buffer (Section 2.2.2.1).

Throughput

Figure 4-1 shows the variation of total throughput, η, with total offered load, G, for P = 64, 128 and 512 bytes. For G less than 80-90% virtually no collisions occur and η is equal to G. Thereafter, packets begin to experience collisions and η levels off to some value directly related to P after reaching a peak, η_{max}. For short packets, $P = 64$, this maximum is about 80%, for longer packets, $P = 512$, it is above 95%. The network remains stable even under conditions of heavy overload owing to the load-regulation of the back-off algorithm. (These curves are similar to the ones obtained by Shoch and Hupp [Shoch & Hupp 80]).

Delay

Figure 4-2 shows the delay-throughput performance for the same set of packet lengths. In each case, for η less than η_{max}, the delay is approximately equal to the packet transmission time, i.e., there is almost no contention delay for access to the network. As the throughput approaches the maximum, the delay rises rapidly to several times the packet transmission time owing to collisions and the associated back-offs.

Figure 4-3 shows the histograms of the cumulative delay distributions for low, medium and high offered loads for P = 64, 128, 512 bytes. The delay bins are logarithmic. The labels on the X-axis indicate the upper limit of each bin. The left-most bin includes packets with delay ≤ 0.57 ms, the next bin, packets with delay ≤ 1.18 ms, and so on. The ordinate is the number of packets expressed as a percentage of all successfully

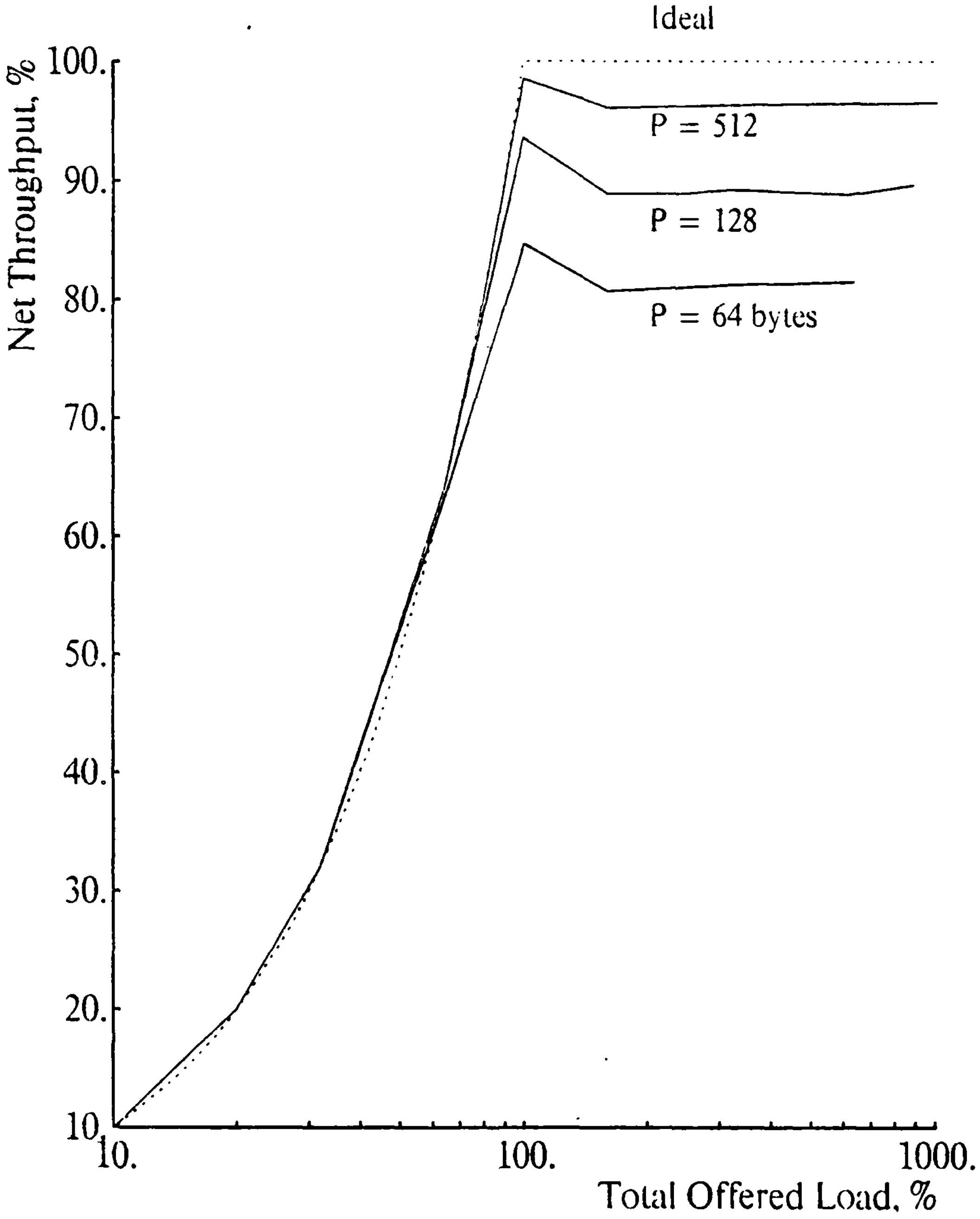


Figure 4-1: 3 Mb/s Ethernet: Throughput vs. Offered load. Measurements. 32 stations. Parameter, *P*.

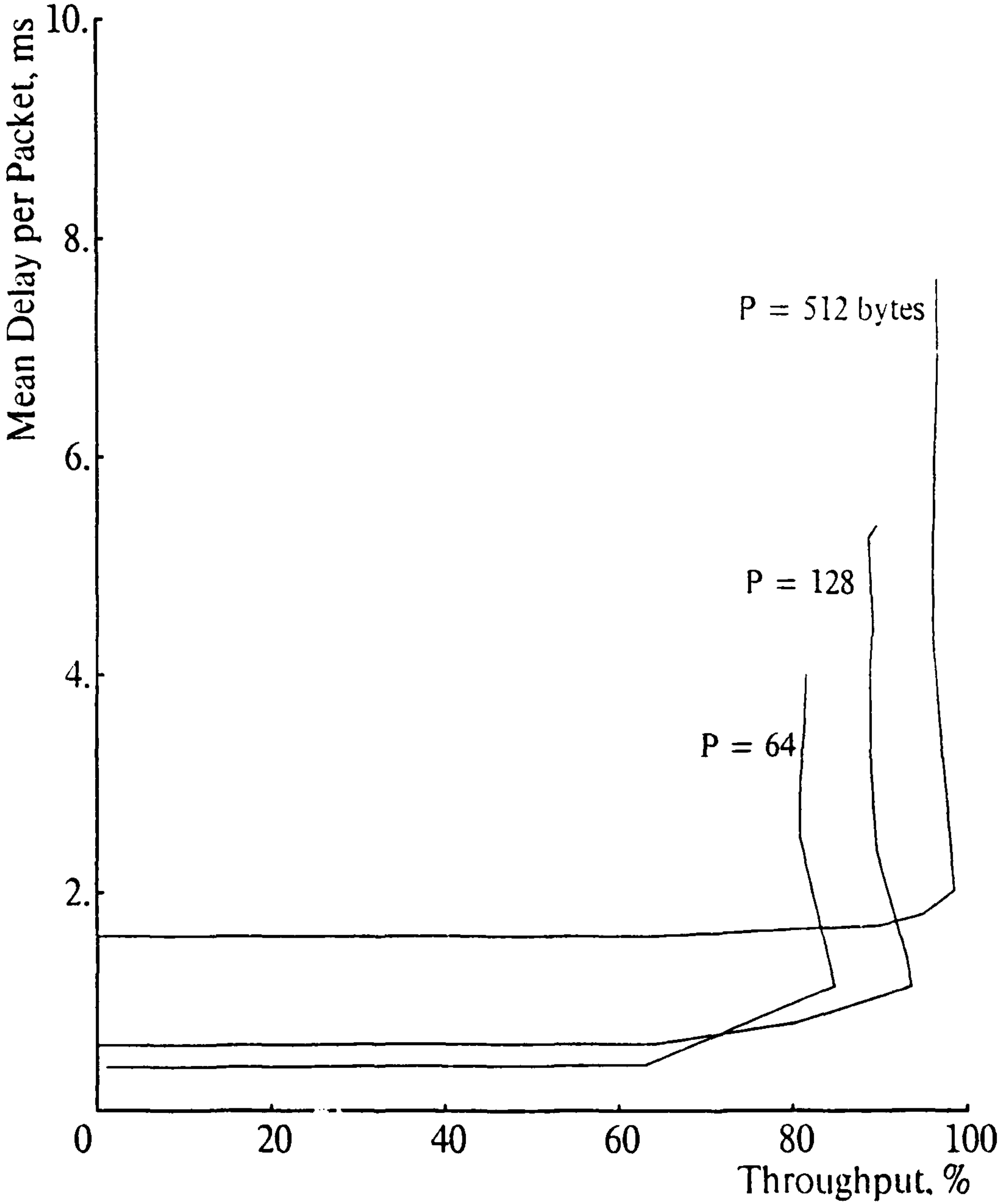


Figure 4-2: 3 Mb/s Ethernet: Delay vs. Throughput. Measurements. 32 stations. Parameter, *P*.

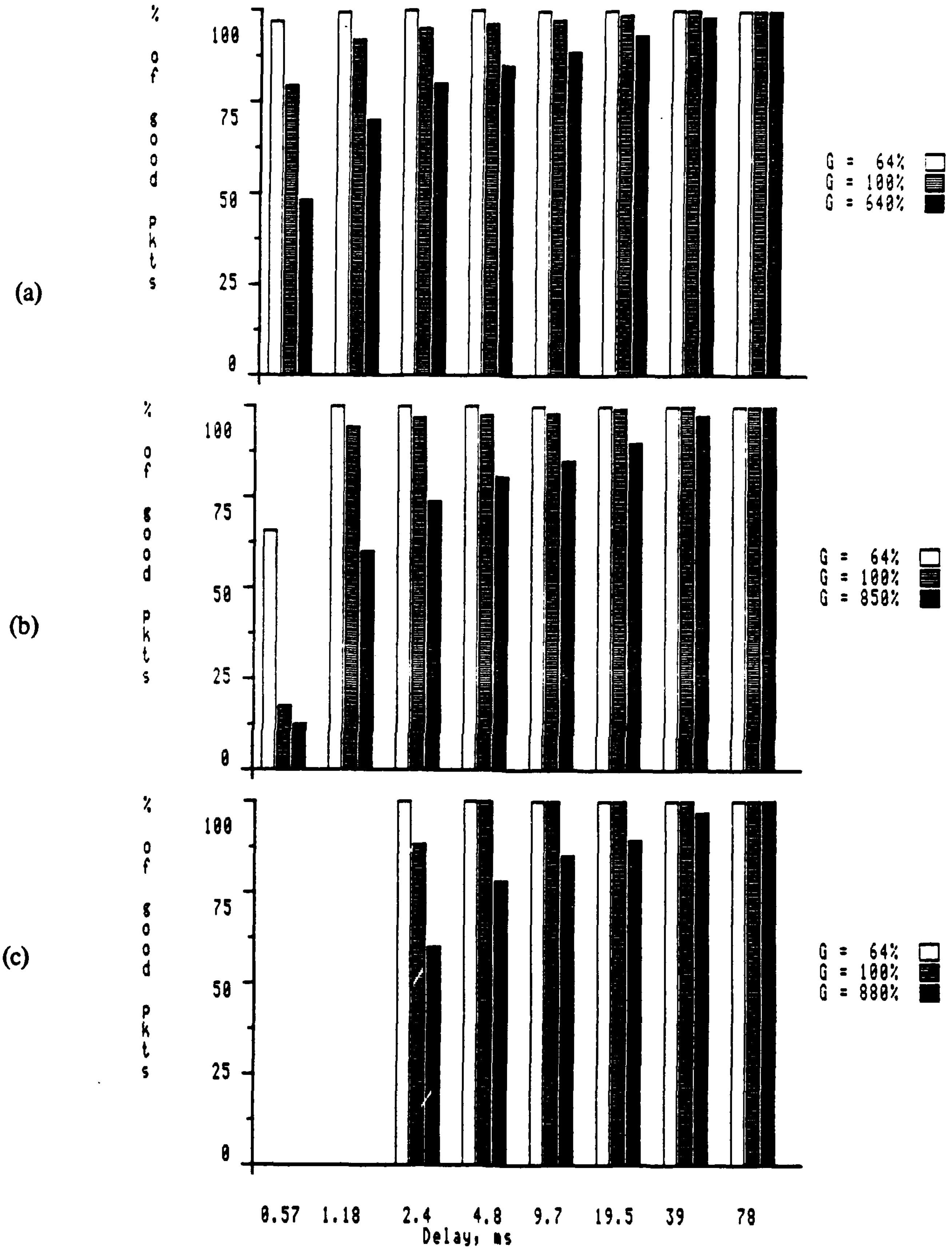


Figure 4-3: 3 Mb/s Ethernet: Cumulative Delay Distribution. Measurements. 32 stations.
(a) P = 64 bytes. (b) P = 128 bytes. (c)-P = 512 bytes.

transmitted packets. Table 4-1 gives the fraction of the total packets generated that were successfully transmitted.

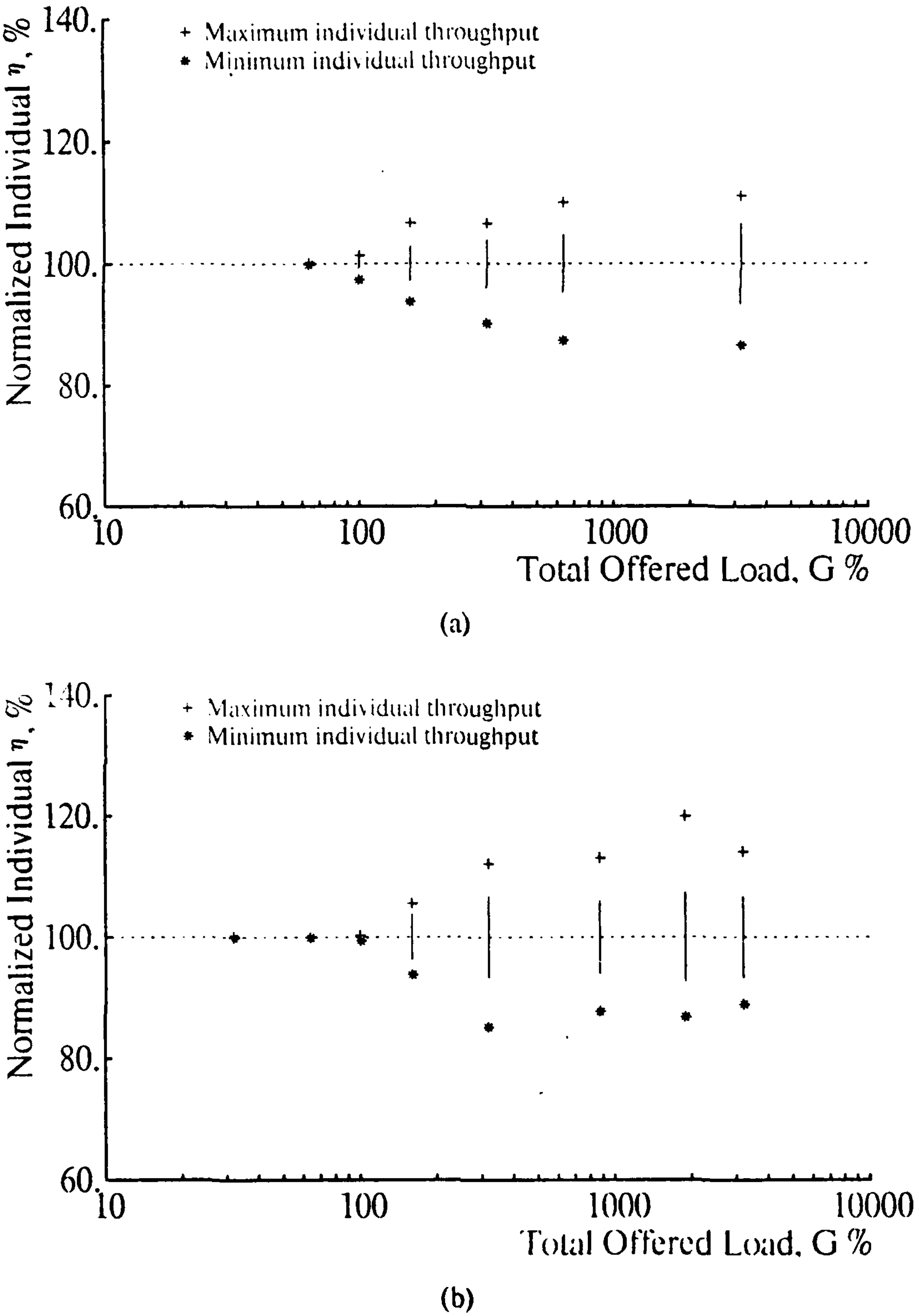
P, bytes	*G*, %	Successful Packets / Total Packets %
64	64	100.0
	100	99.8
	640	96.1
128	64	100.0
	100	99.7
	850	88.3
512	64	100.0
	100	99.9
	880	58.2

Table 4-1: 3 Mb/s Ethernet: Successfully transmitted packets as a percentage of total packets

We see that for $G = 64\%$, the delay of all packets is approximately the minimum, i.e., there is little queueing for network access. Even with $G = 100\%$ most packets suffer delays of less than 5 ms. Under heavy load conditions, however, only about 75% of the packets have delays of less than 5 ms, with the remainder suffering delays of up to 80 ms.

Fairness

To investigate the fairness of the protocol to contending stations, we examine variations in performance metrics measured by individual stations with increase in *G*. In Figure 4-4 we plot the normalized mean of the individual throughputs vs. *G* for $P = 64$ and 512 bytes. The vertical bars indicate the normalized standard deviation, i.e., the coefficient of variation. Also shown are the maximum and minimum individual throughputs. For low *G*, there is little variation in individual throughput. Under overload, the variation increases but remains less than ±10%.



(a)

(b)

Figure 4-4: 3 Mb/s Ethernet: Variation in η per station vs. G. Measurements. 32 stations. (a) P = 64 bytes. (b) P = 512 bytes.

Figure 4-5 contains similar plots for the mean delay per packet measured by each station. The variations with G are seen to be similar to those in Figure 4-4. Thus the protocol appears to be fair to all contenders. (This has been noted in other experiments [Shoch & Hupp 80]). In Section 4.4 we show that the variations seen in station metrics are not purely random but are dependant in part on the physical location of the stations on the network. Further, the bias due to location is greater with larger a, where a is the ratio of end-to-end propagation delay, τ_p, to the packet transmission time, T_p.

The metrics show slightly higher variation for $P = 512$ than for $P = 64$ bytes. During the successful transmission of a 512 byte packet a larger number of stations are likely to queue for access to the network. Thus, at the end of the transmission all these stations will attempt to transmit and will collide. The time for resolution of the collision is dependent on the number of colliding stations and hence may be expected to be longer for $P = 512$ than for $P = 64$.

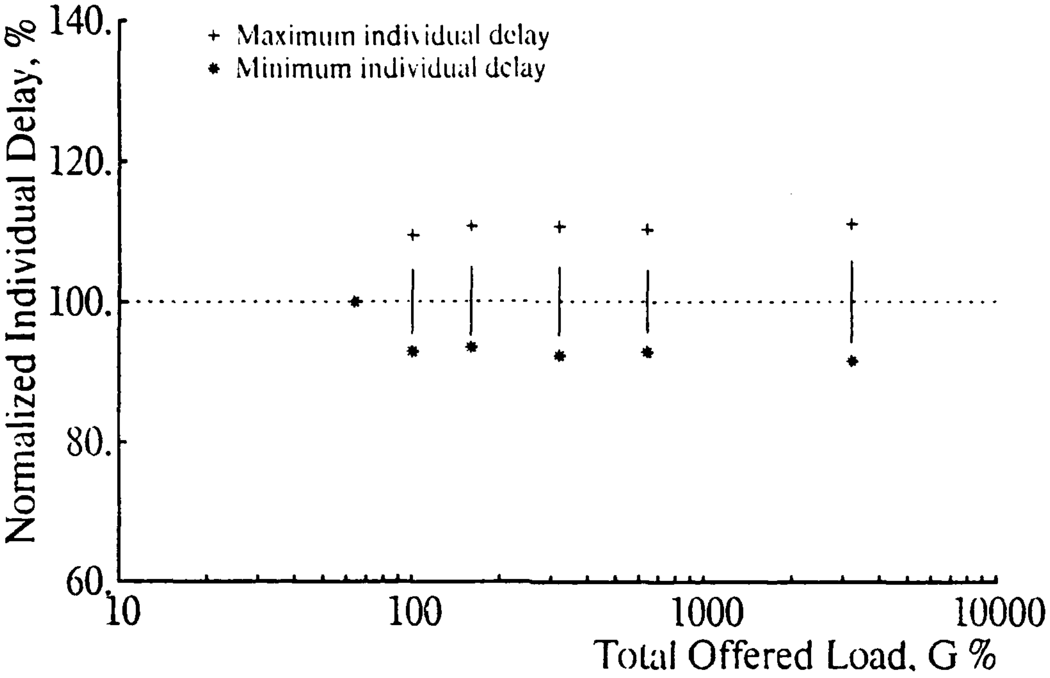
4.2.3. 10 Mb/s Ethernet

We now consider the performance of a 10 Mb/s Ethernet network. The results presented were obtained using 30 - 38 transmitting stations.[8] As in the 3 Mb/s case, fixed length packets were used with the inter-arrival times of packets at each station being uniformly distributed random variables. Stations were operated in the feedback mode with a single buffer (Section 2.2.2.1).

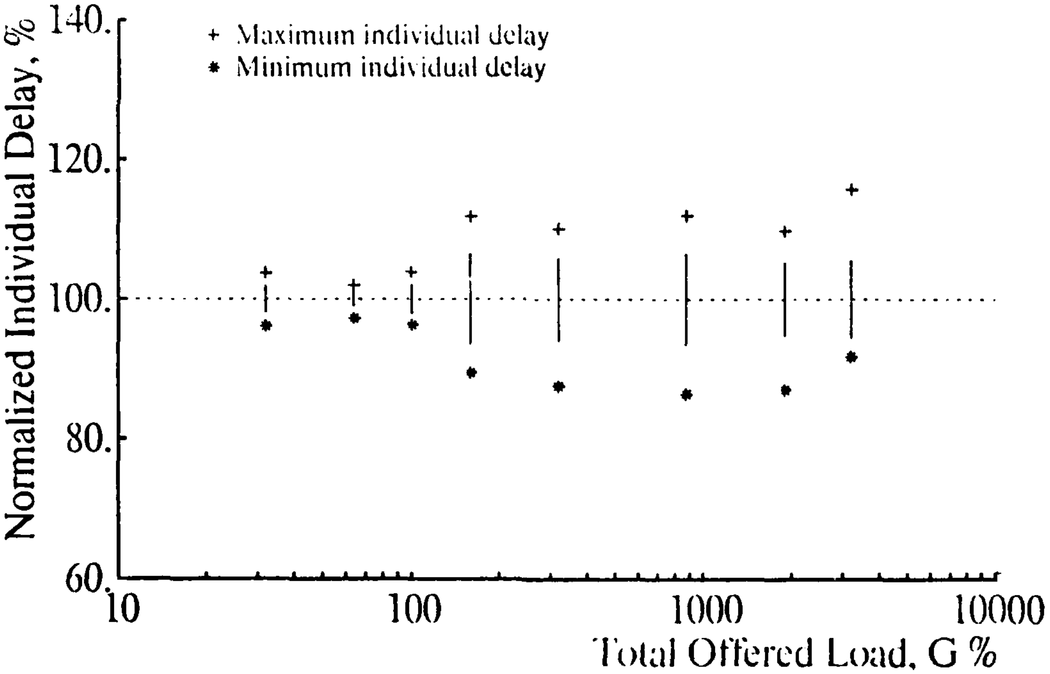
Throughput

Figure 4-6 shows the throughput as a function of total offered load, G, for P ranging from 64 to 5000 bytes. The shape of the curves is similar to the corresponding curves for the 3 Mb/s Ethernet. However, maximum throughput varies from 25% for $P = 64$ bytes (the minimum allowed by the Ethernet specifications [Ethernet 80]), to 80% for $P = 1500$ bytes (the maximum allowed), to 94% for very long packets of 5000 bytes. We note that for each curve, for G below the knee point, the throughput is approximately equal to G. Even under conditions of heavy overload, the network remains stable.

[8] The number of stations for a given set of experiments, i.e., for a single packet length, was constant.



(a)



(b)

Figure 4-5: 3 Mb/s Ethernet: Variation in delay per station vs. *G*. Measurements. 32 stations.
(a) *P* = 64 bytes. (b) *P* = 512 bytes.

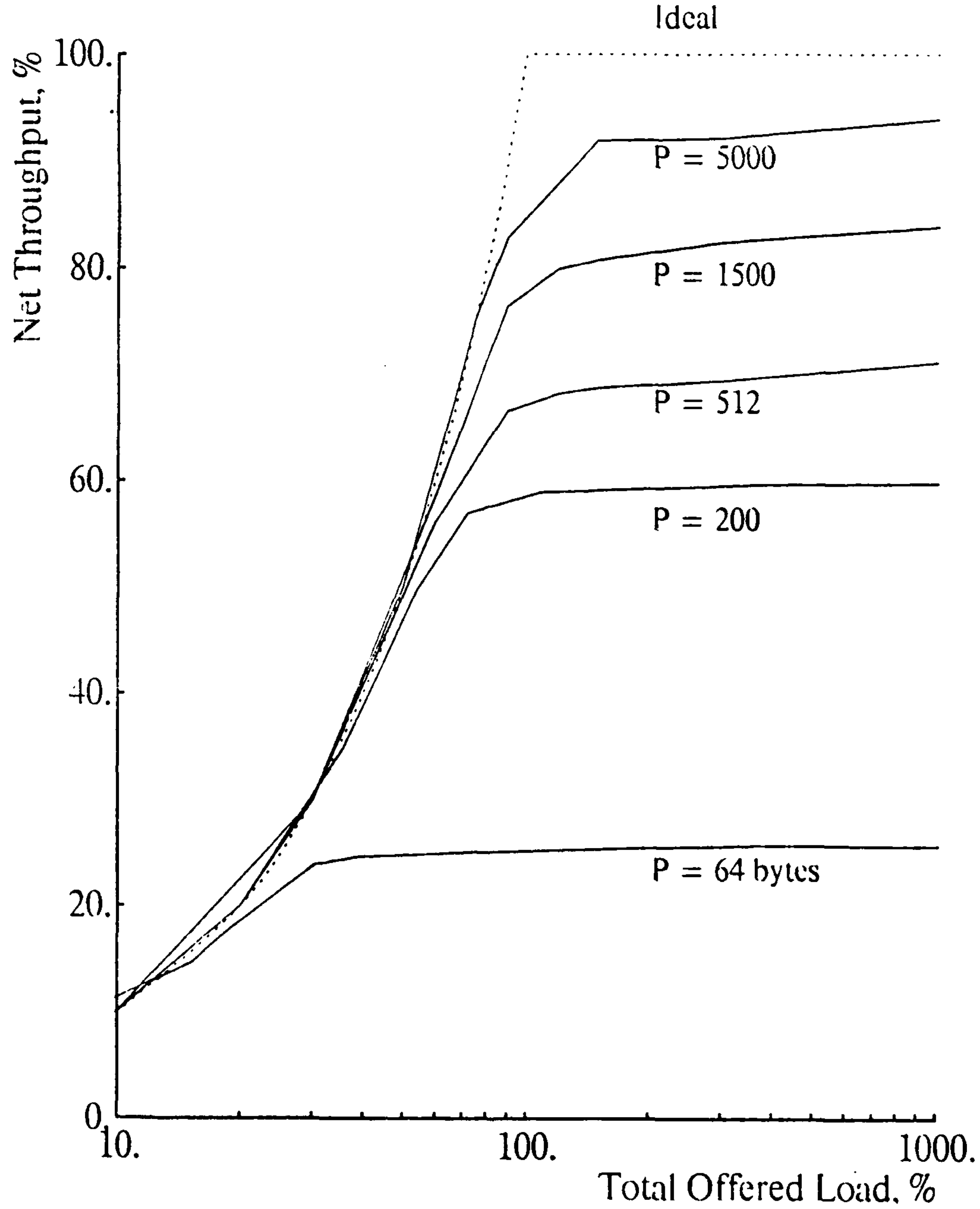


Figure 4-6: 10 Mb/s Ethernet: Throughput vs. Offered load. Measurements. 30 - 38 stations. Parameter, *P*.

Delay

Figure 4-7 shows the delay-throughput performance for the same set of packet lengths. Again, the curves have similar shapes to the curves for the 3 Mb/s network. For G below the knee points, the delay is minimal, whilst above the knee points, it rises sharply. The knees in this case are less pronounced, especially for larger P.

Figure 4-8 shows the histograms of the cumulative delay distributions for low, medium and high offered loads for $P = 64$, 512 and 1500 bytes. Table 4-2 gives the fraction of the total packets generated that were successfully transmitted. For low loads, delay is minimal for $P = 512$ and 1500 bytes. For $P = 64$, though, even at $G = 19\%$ delays range up to $10T_p$. At high loads, for all packet lengths, the majority of packets suffer moderately increased delays, while a fraction suffer very high delays, up to 0.5s. for $P = 512$ and 1500, about 75% or all packets suffer delays $\leq 10T_p$. For $P = 64$, 75% of packets suffer delays $\leq 15T_p$.

Fairness

We examine variations in performance metrics measured by individual stations with increase in G. In Figure 4-9 and 4-10 we plot the normalized means of the individual throughputs and delays respectively vs. G for $P = 64$ and 512 bytes. The vertical bars indicate the normalized standard deviation, i.e., the coefficient of variation. Also shown are the maximum and minimum individual throughputs.

The metrics show higher variation than in the 3 Mb/s case, in the range ±35%. This may be attributed to the larger retransmission periods in the 10 Mb/s Ethernet back-off algorithm (see Sections 4.1.2 and 4.1.3). Also, the dependence on G is less marked. Contrary to the 3 Mb/s case, the variation is slightly lower here for the larger packets size. The issue of fairness is considered further in Section 4.4.

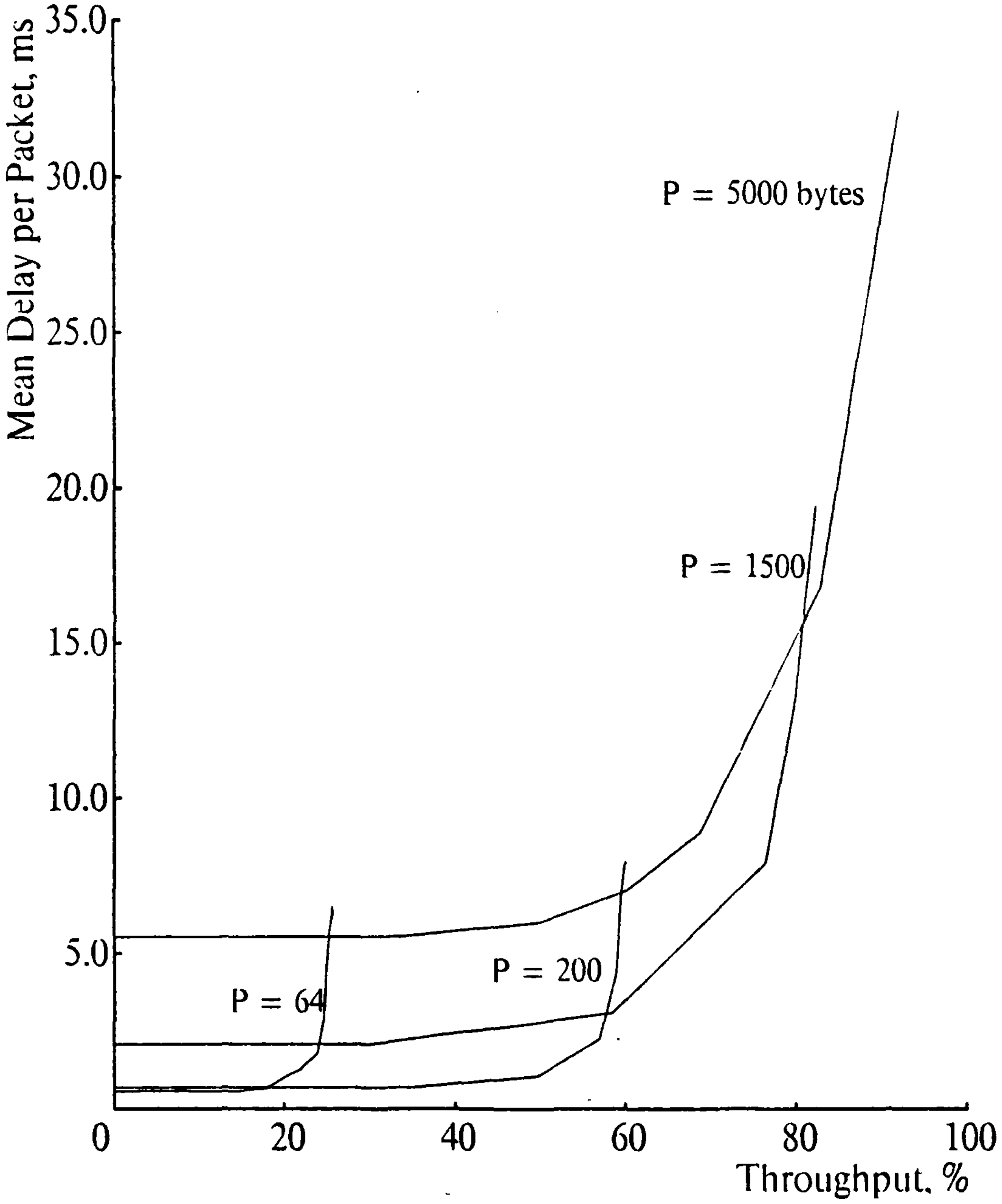


Figure 4-7: 10 Mb/s Ethernet: Delay vs. Throughput. Measurements. 30 - 38 stations. Parameter, *P*.

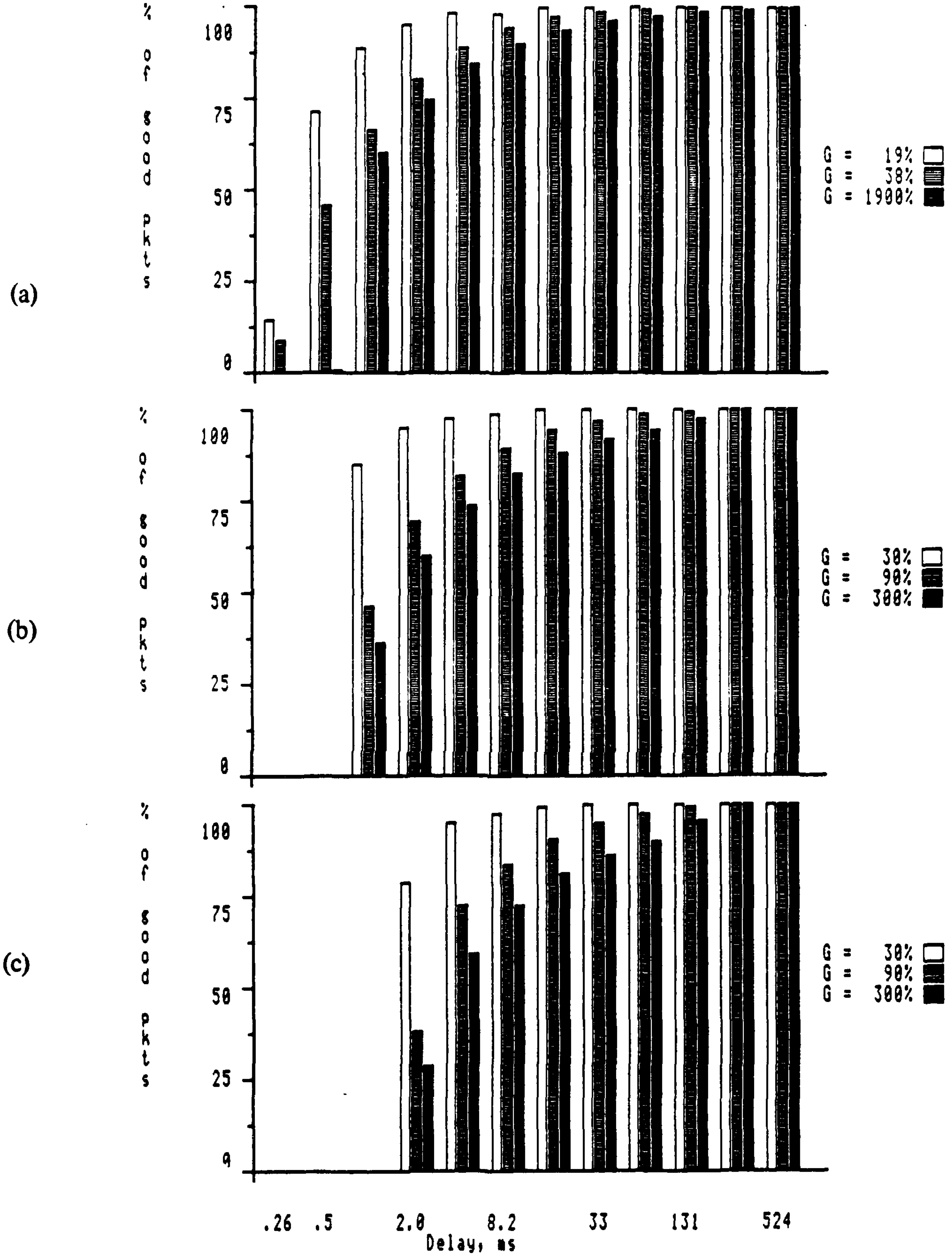
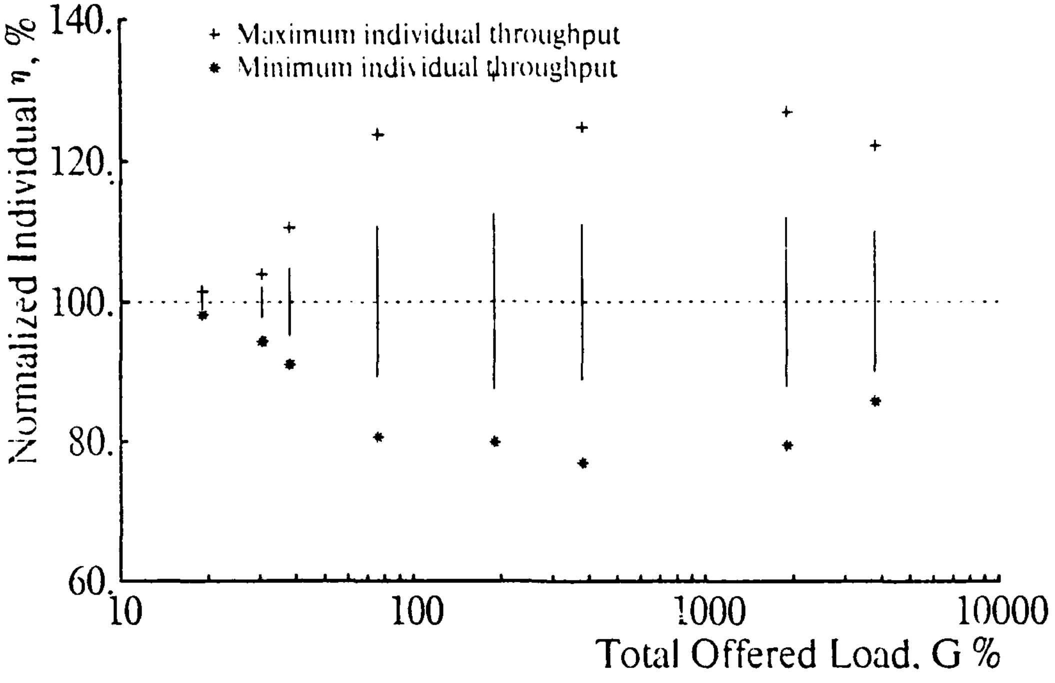


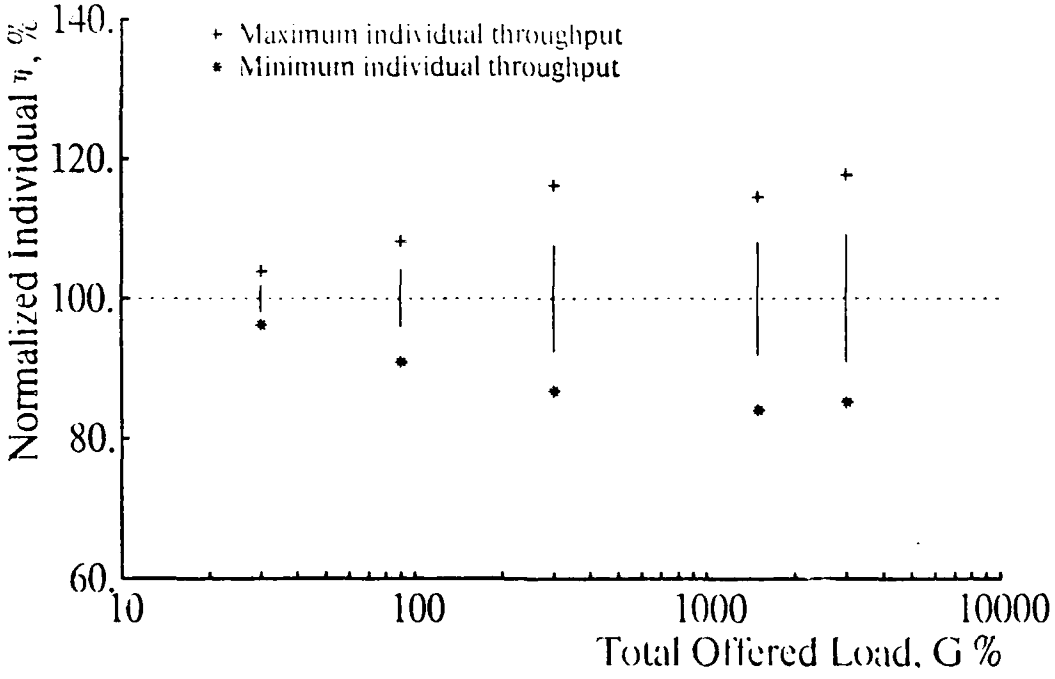
Figure 4-8: 10 Mb/s Ethernet: Cumulative Delay Distribution. Measurements. 38 stations. $P = 64$ bytes.)
(a) $P = 64$ bytes. (b) $P = 512$ bytes. (c) $P = 1500$ bytes.

P, bytes	*G*, %	$\frac{\text{Successful Packets}}{\text{Total Packets}}$ %
64	19	100.0
	38	100.0
	1900	99.9
512	30	100.0
	90	99.9
	300	98.7
1500	30	100.0
	90	99.7
	300	93.3

Table 4-2: 10 Mb/s Ethernet: Successfully transmitted packets as a fraction of total packets

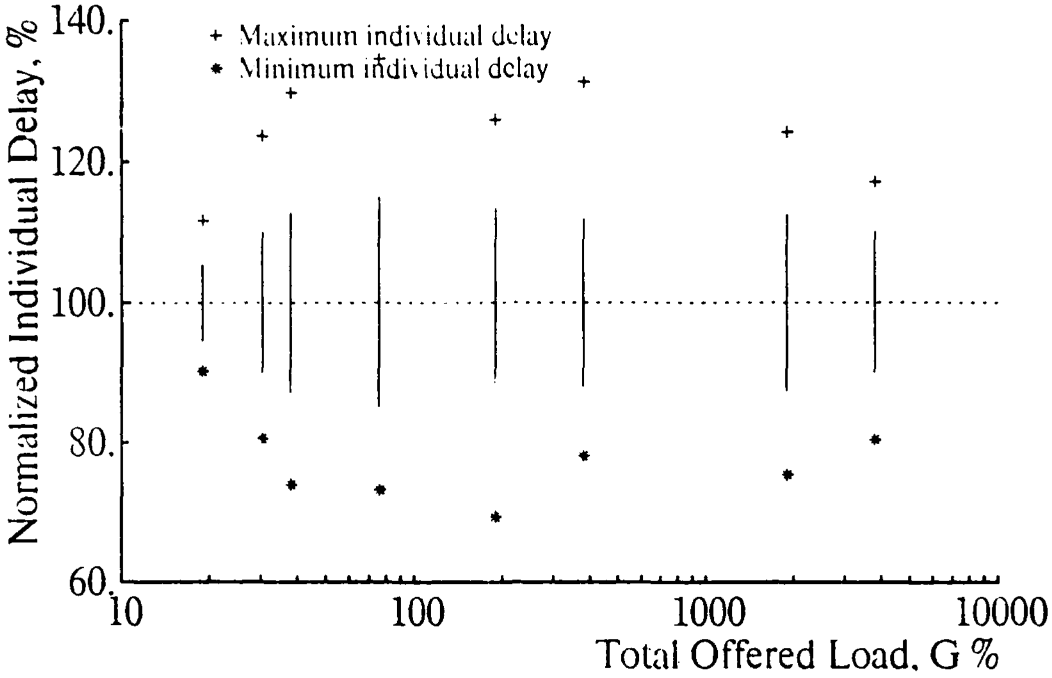


(a)

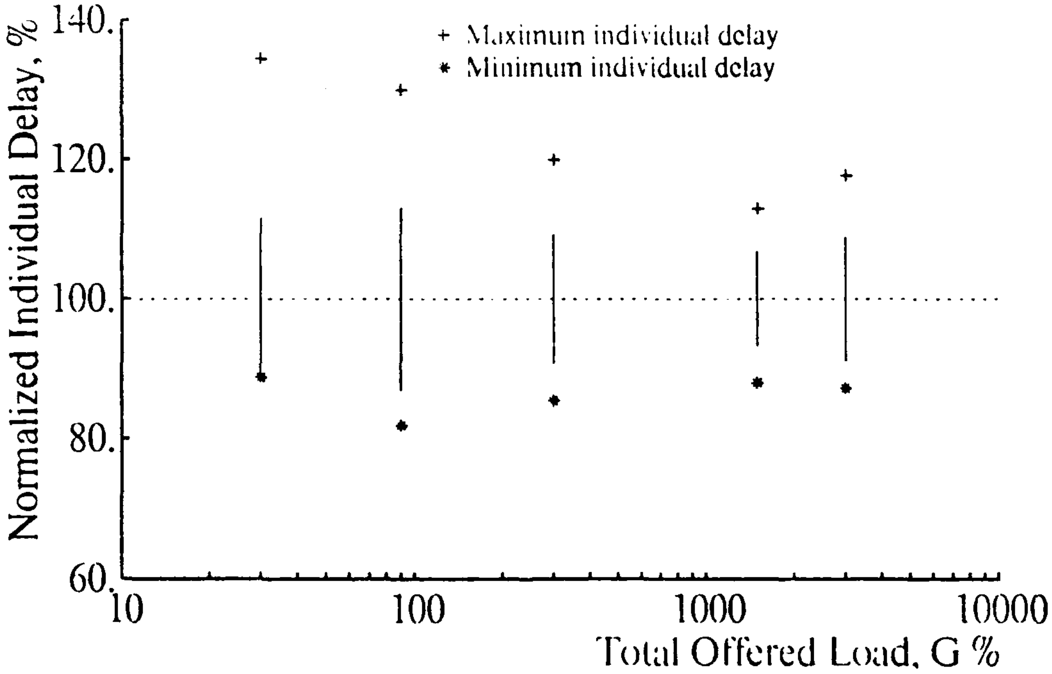


(b)

Figure 4-9: 10 Mb/s Ethernet: Variation in η per station vs. G. Measurements. 30 - 38 stations. (a) P = 64 bytes. (b) P = 512 bytes.



(a)



(b)

Figure 4-10: 10 Mb/s Ethernet: Variation in delay per station vs. *G*. Measurements. 30 - 38 stations. (a) P = 64 bytes. (b) P = 512 bytes.

4.2.4. Comparison of the 3 and 10 Mb/s Ethernets

In this section we examine the effect of the difference in bandwidth between the two Ethernets on performance. The differences in some important parameters, such as network length, in the two cases should be borne in mind.

Throughput

Figure 4-11 shows the throughput, η, as a function of total offered load, G, for several packet lengths for the 2 networks. For $P = 64$ bytes, the throughputs of the two networks are almost equal. For longer packets, the 10 Mb/s network exhibits substantially higher throughput. The throughput increases less than linearly with increase in bandwidth. This is shown in Table 4-3 in which the ratio of the absolute throughput at 10 Mb/s to that at 3 Mb/s is given for several values of P.

P, bytes	Ratio of throughputs η_{10}/η_3
64	1.05
512	2.45
1500	2.90

Table 4-3: Increase in η with increase in C from 3 to 10 Mb/s

Delay

Figure 4-12 shows mean packet delay as a function of total offered load, in Mb/s, for the two networks and several values of P. The shapes of all the curves are similar: there is a region of minimal delay at low G, then there is a rapid increase in delay to some saturation value at high G. For low G, the delay is lower in the 10 Mb/s network than in the 3 Mb/s one for a given P. However, in the region of overload, the 3 Mb/s network exhibits lower delay. This is due to the more severe back-off algorithm of the 10 Mb/s network (Sections 4.1.2 and 4.1.3). A comparison with analytical models from the literature is deferred to Section 4.3.2.

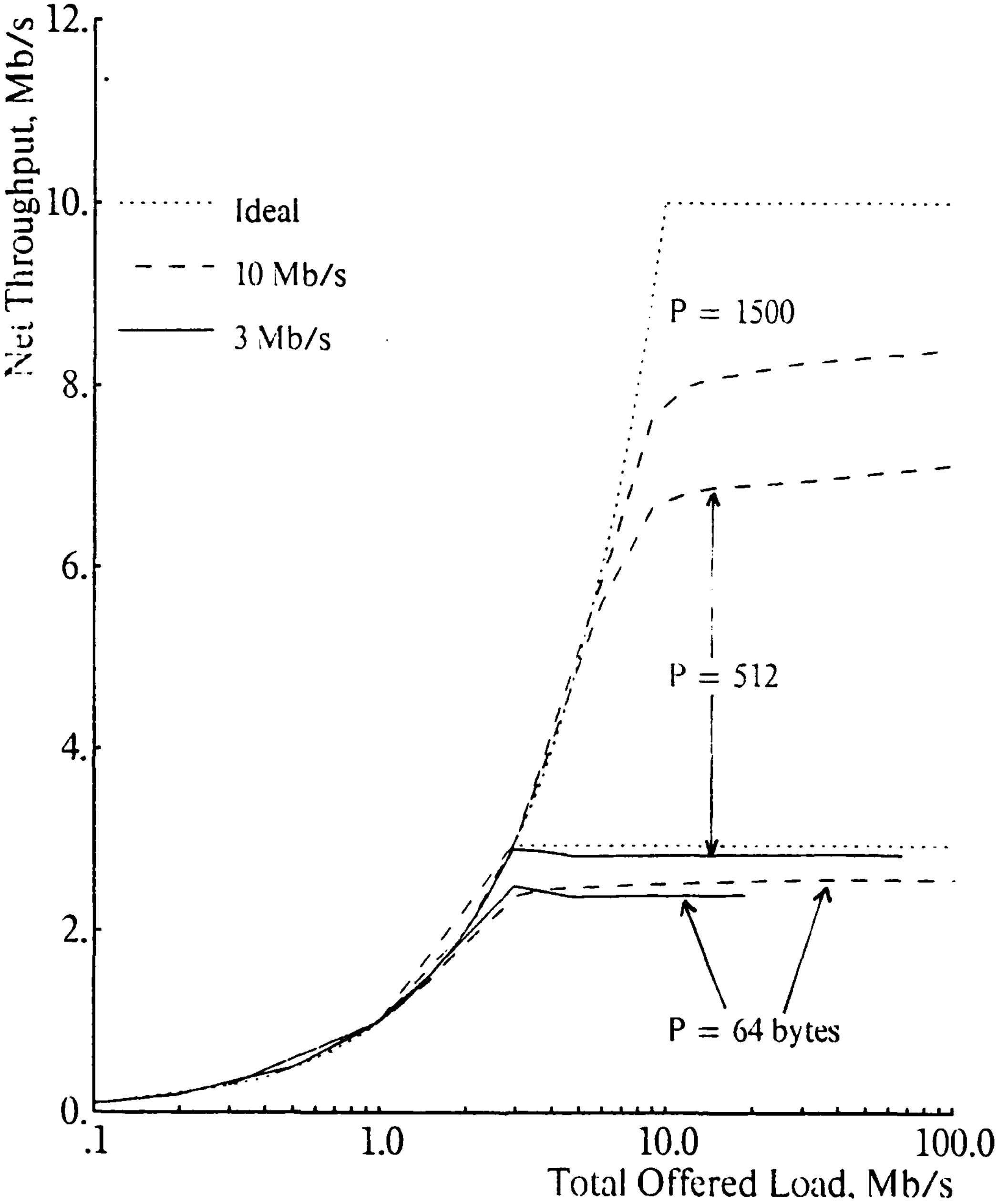


Figure 4-11: 3 & 10 Mb/s Ethernets: Throughput vs. *G*.
Measurements. 30 - 38 stations. Parameter, *P*.

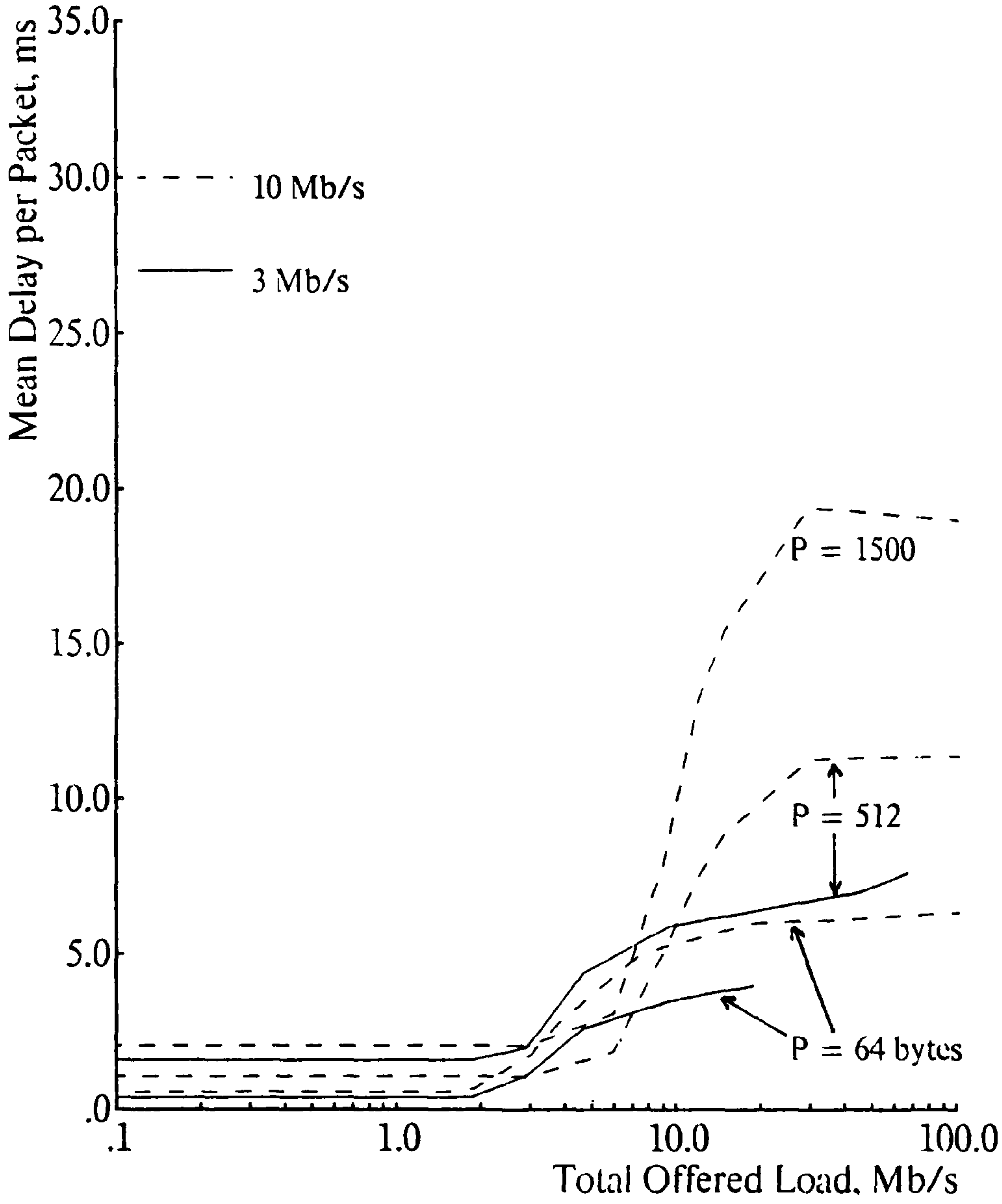


Figure 4-12: 3 & 10 Mb/s Ethernets: Delay vs. *G*. Measurements. 30 - 38 stations. Parameter, *P*.

4.2.5. Discussion

We have used actual measurements on 3 and 10 Mb/s Ethernets with artificially-generated traffic loads to characterize the performance of the protocol under a range of conditions. These include various packet lengths and offered loads ranging from a small fraction of network bandwidth to heavy overload. These experiments span the range from the region of high performance of CSMA/CD networks to the limits at which performance begins to degrade seriously. The former occurs when the packet transmission time is large compared to the round-trip propagation delay. The latter occurs when the two times are comparable in magnitude.

The 3 Mb/s Ethernet is found to achieve utilizations of 80 - 95% for the range of packet lengths considered, 64 - 512 bytes. At a bandwidth of 10 Mb/s, with short packets, e.g., 64 bytes, T_p becomes comparable to τ_p and hence the maximum throughput is low as a percentage of bandwidth. With long packets, greater than 500 bytes in length, high throughputs are achieved. Thus, packet lengths on the order of 64 bytes on a 10 Mb/s network approach the limit of utility of the Ethernet protocol. This does not imply that such short packets should not be used. A study of traffic on a typical local computer network shows that approximately 20% of the total traffic is composed of short packets, whilst the remainder of 80% consists of long packets [Shoch & Hupp 80]. The 10 Mb/s Ethernet studied could support a high throughput with such a traffic mix although it cannot do so with only short packets. Both versions of the protocol are found to be stable under overload, i.e., throughput under overload tends to a saturation value that is equal to or marginally lower than the maximum throughput. Individual stations achieve similar, though not identical, performance. The 10 Mb/s Ethernet exhibits somewhat higher variation in individual station performance than the 3 Mb/s network.

For $G <$ 100%, i.e., most practical situations, the delay is within a small multiple of the packet transmission time, T_p. Under such conditions, the network could provide satisfactory service to real-time traffic with delay constraints. However, at heavy load, delays are as high as 80 ms and 500 ms in the 3 and 10 Mb/s networks respectively. The majority of packets still suffer relatively minimal delays. While a station is incurring a back-off delay, it is not contending for network access. Thus, large delays effectively reduce the instantaneous offered load and help maintain stability.

Thus, the Ethernet protocol is seen to be very suitable for local interconnection when the packet length can be made sufficiently large and occasionally highly variable delays can be tolerated. This matches the nature and requirements of most computer communication traffic. However, for real-time communications, such as digital voice telephony, the utility of the Ethernet is more restricted. This is explored further in Section 4.5.

4.3. Data Traffic: Measurement, Simulation and Analysis

Since analytic models of the Ethernet are not available, we are interested to see whether models of CSMA/CD from the literature may be used to predict Ethernet performance. This investigation allows us to use the insights gained from such models for the improvement of the Ethernet protocol. As a side-effect, we are able to determine the regions of applicability of such models for the prediction of Ethernet performance. First, the models are briefly described, with the differences from the Ethernet being emphasised. Next, we present a comparison of the analytical predictions to our measured results. This indicates several differences, especially in the region where performance begins to degrade, i.e., when $a = \tau_p/T_p$ is large and/or the number of stations is large. We examine in more detail, via simulation, Ethernet performance under these conditions. We show that a simple modification to the back-off algorithm enables near-optimal throughput to be maintained even with large numbers of stations.

4.3.1. The Analytical Models

While several analytic models of CSMA/CD have appeared (see Section 1.2), we choose three for further study here. Metcalfe & Boggs derived a simple formula for prediction of the capacity of finite-population Ethernets [Metcalfe & Boggs 76]. This is of interest because of its simplicity and because it was presented along with the first published description of the Ethernet implementation. Later, more sophisticated stochastic analyses of delay and throughput appeared. Lam used a single-server queueing model to obtain fairly simple expressions for delay and throughput [Lam 80]. Tobagi & Hunt applied the method of embedded Markov chains [Kleinrock & Tobagi 75] to more accurately model CSMA/CD [Tobagi & Hunt 80]. This study obtained delay characteristics even for the finite-population case but involves greater computational complexity. These models differ from the Ethernet primarily in the retransmission strategy used upon a collision. Other differences include the assumption of slotted operation and the topology as noted below. For complete details of the analyses the reader is referred to the papers cited above.

Metcalfe & Boggs, 1976

Metcalfe & Boggs derive a simple formula for η_{max} with a finite population of stations, N, and fixed packet length, P [Metcalfe & Boggs 76]. It is attractive because of its computational simplicity. The assumptions necessary to achieve this, however, differ from the Ethernet implementation. The topology is assumed to be a balanced star, i.e., every pair of stations is separated by the same distance (Figure 4-13). The channel is assumed to be slotted in time. Packets arriving during a slot wait until the start of the next slot at which time all ready stations simultaneously begin transmission. The slot duration is chosen to at least $2\tau_p$ so that any collision can be detected and transmission aborted within one slot. This slotting lumps together two independent parameters, τ_p and t_{jam}, the jam time. This is especially poor if $t_{jam} \gg \tau_p$. Packet arrivals, both new and retransmissions, at each station are assumed to be such that each station attempts to transmit with the optimum probability of $1/N$ in each slot. The Ethernet implementation attempts to approximate this behaviour. With N stations attempting to transmit, the probability of a success, A, is the probability that exactly one chooses to transmit in the slot. Thus,

$$W = \frac{1-A}{A}$$

Thus, we have the efficiency,

$$\eta = \frac{P/C}{P/C + W\cdot 2\tau_p}$$
$$= \frac{1}{1 + W\cdot 2\tau_p/T_p}$$
$$= \frac{1}{1+2aW}$$

where C is the channel bandwidth, T_p the packet transmission time and $a = \tau_p/T_p$.

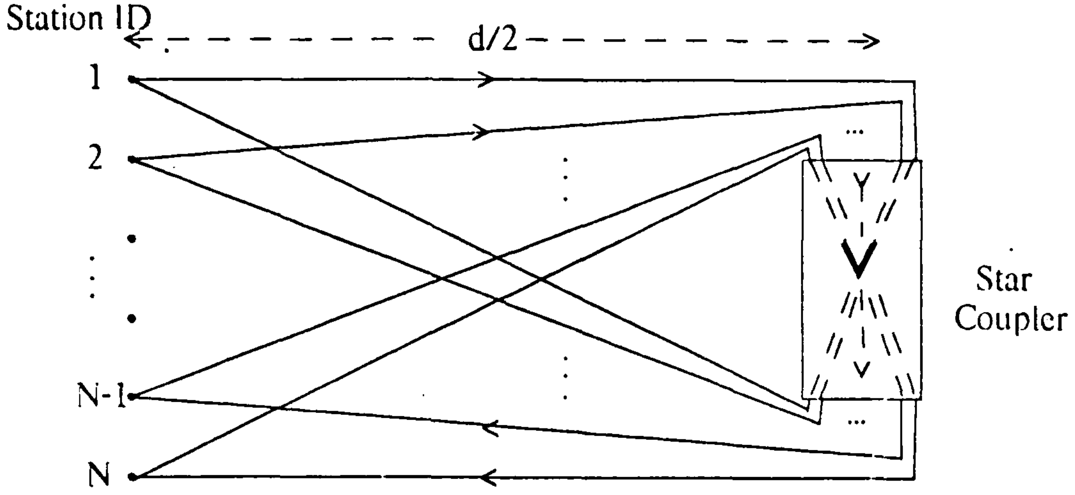


Figure 4-13: Ethernet Topology: Balanced Star

Tobagi & Hunt, 1980

Tobagi & Hunt first present a model for estimating the maximum throughput of a CSMA/CD network under the assumption of an infinite population [Tobagi & Hunt 80]. The propagation delay between every pair of stations is assumed to be τ_p, i.e., the topology is the balanced star. The channel is assumed to be slotted in time, with slot duration τ_p, to permit the formulation of a discrete-time model. In contrast to Metcalfe & Boggs, t_{jam} is not included in the slot time but, more accurately, is assumed to be independent. Retransmission delays are assumed to be arbitrarily large since the aim is to obtain the maximum throughput. Hence the arrival of new and retransmitted packets can be assumed to be independent and to form a Poisson process.

Next, a delay-throughput analysis is presented for a finite population system.[9] The retransmission delay after a collision is assumed to be a geometrically distributed random variable with mean $1/\nu$, with constant ν, independent of the packet transmission in question and the number of collisions incurred so far. For a given set of parameters, the optimum delay-throughput performance is obtained by plotting the delay-throughput

[9] The analysis for the 0-persistent case is presented [Tobagi & Hunt 80]. This is extended to the 1-persistent case by [Shacham & Hunt 82].

curves for various values of ν and taking their lower envelope. These optimum curves have D increasing monotonically with η. As $1/\nu$ becomes large, η tends to an asymptotic value and D becomes large. The asymptotic throughput is approached for a value of ν that decreases with N. It is shown there that for finite N, CSMA/CD exhibits a stable behaviour even when the rescheduling delay has a constant mean. When this mean is chosen optimally as a function of N, high throughput is achieved. As $N \rightarrow \infty$, $\nu \rightarrow 0$, i.e., retransmission delays become arbitrarily large. For sufficiently large populations, e.g., 50 stations at $a = 0.01$, the asymptotic throughput approaches η_{max} derived in the infinite population analysis, relatively insensitive to N. Thus, in the limit the throughput predictions of the finite and infinite population analyses converge and so we use the computationally simpler infinite population analysis to obtain η_{max} for our comparisons.[10]

Lam, 1980

Lam uses a single-server queueing model to approximate the distributed protocol of the Ethernet [Lam 80]. As such it is unable to capture many of the details of the protocol though it is attractive as the expressions obtained are computationally simpler than those of the more exact analysis described above. New packets are assumed to arrive from an infinite population of users in a Poisson process. The balanced star topology is assumed. As in the Metcalfe & Boggs model, the channel is assumed to be slotted with the slot duration at least $2\tau_p$. Here too the slot duration is assumed to be $t_{jam}+2\times\tau_p$, leading to inaccuracy when $t_{jam} \gg \tau_p$. The access protocol differs from the Ethernet in the behaviour after a collision. The retransmission algorithm is assumed to be such that the probability of a successful transmission in each slot is $1/e$. The mean number of slots from the end of the first collision until the next successful transmission is geometrically distributed with mean $(e-1)$. This optimal retransmission algorithm requires that full knowledge of the state of the system be instantaneously available at all stations. Owing to the constant probability of a success, the system is stable and delay-throughput curves are similar in shape to the optimum curves obtained by Tobagi & Hunt. The asymptotic throughput, η_{max}, is the value we use in the comparison.

[10] An earlier analysis of CSMA considered dynamic ν to help minimize delay and maximize capacity [Tobagi & Kleinrock 77]. For large N, it was found that capacity did not exceed that of the corresponding infinite-population analysis.

4.3.2. Measurement and Analysis: Comparison

We now compare the predictions of the models described in the previous Section to our measurements (Section 4.2). First we consider the maximum throughput. From the formula of Metcalfe & Boggs we compute η with the number of hosts, N, equal to 32 to correspond to our measurements (the predicted η does not vary much with N). We use the infinite population analysis of Tobagi & Hunt to obtain η_{max} from the formula for η as a function of G. Table 4-4 shows measured and computed values of maximum throughput for various values of P for the 3 Mb/s Ethernet. τ_p is estimated to be 3 μs (see Section 4.1.2). Table 4-5 and 4-6 show corresponding sets of values for the 10 Mb/s Ethernet. The measured values in Table Ref(TabEthCfMeasAnal0Ms) were obtained on a configuration consisting of 750 m of cable with 1 repeater. The measured values in Table 4-6 were obtained on the configuration described in Section 4.1.3. τ_p is estimated at 11.75 and 15 μs respectively.[11]

P bytes	a	Measured	Metcalfe & Boggs, 80	Tobagi & Hunt, 80	Lam, 80
64	0.016	82	87	80	80
128	0.008	89	93	89	89
512	0.002	97	98	97	97

Table 4-4: 3 Mb/s Ethernet: Maximum Throughput, %
τ_p = 3 μs (550 m)

[11] a is computed using the total packet length, including overhead. Throughputs shown are net, excluding overhead.

P bytes	a	Measured	Metcalfe & Boggs, 76	Tobagi & Hunt, 80	Lam, 80
64	0.22	26	55	40	28
200	0.072	62	79	60	54
512	0.028	72	91	77	75
1500	0.0098	86	97	91	90
5000	0.0029	95	99	97	97

Table 4-5: 10 Mb/s Ethernet: Maximum Throughput. %)
$\tau_p = 11.75\ \mu s$ (750 m + 1 repeater)

P bytes	a	Measured	Metcalfe & Boggs, 76	Tobagi & Hunt, 80	Lam, 80
64	0.28	26	49	37	25
200	0.092	60	75	58	51
512	0.036	72	88	74	72
1500	0.012	85	96	89	89
5000	0.0037	94	99	96	96
10000	0.0019	97	99	98	98

Table 4-6: 10 Mb/s Ethernet: Maximum Throughput. %)
$\tau_p = 15\ \mu s$ (1500 m + 2 repeaters)

It is seen that the simple formula of Metcalfe & Boggs overestimates η_{max}, with the error being small for $a < 0.01$ but as high as +100% at large values of a. This may be attributed to the assumption of an optimum retransmission policy. The assumption of slotted operation is also expected to lead to higher predicted capacity. On the other hand, the assumption that every pair of hosts is separated by τ_p would lower the predicted throughput, but appears to be less significant than the other assumptions. Note that with larger t_{jam}, the value of a at which the error becomes large would be lower because t_{jam} is lumped into the slot duration.

The model of Tobagi & Hunt provides better estimates of η for $a < 0.1$. However, the correspondence is inconsistent, especially at large a. The primary difference between the model and the Ethernet is the nature of the retransmission algorithm. The inconsistency may be due to the accuracy of the estimation of τ_p and to the opposing effects of two assumptions: the optimistic slotted assumption and the pessimistic balanced star topology assumption. Finally, the comparison of infinite population analysis to finite population measurements may be misleading. These issues are addressed in Section 4.3.3.

Considering Lam's model, it is seen to provide fairly good estimates of throughput. However, the correspondence is strongly dependant on the particular parameters under which the measurements were conducted. As we will see in Section 4.3.3, the correspondence is poor for other sets of parameters yielding the same value of a. This is due to the approximations introduced in modelling CSMA/CD by a simple single-server queueing model. We note that as in Metcalfe & Boggs' model the validity of Lam's model is further limited by the lumping of t_{jam} and τ_p.

4.3.3. Further Exploration via Simulation

We use simulation, validated against the measurements presented earlier (see Appendix B.2), to explore further the performance of the Ethernet in the regions in which the analytic models are poor predictors. First, we consider the effects of the number of stations, N, and the number of buffers per station, B. Next, based on the theoretical predictions that for a given N stable behaviour can be achieved by using a fixed retransmission delay, dependant on N, we propose and study a modification to the Ethernet algorithm [Tobagi & Kleinrock 77, Tobagi & Hunt 80]. Finally, we compare the star and linear bus topologies. The understanding gained in these exercises enables us to determine the applicability of the analytical models to the prediction of Ethernet performance.

The simulation parameters are chosen to resemble the 10 Mb/s Ethernet. The signal propagation delay is assumed to be 0.005 μs/m. The end-to-end propagation delay, τ_p, is assumed to be 10 μs. This corresponds to a channel length, d, of 2000 m using a single cable segment, or to a lower value if repeaters are used. The star topology (Figure 4-13) is assumed, except where otherwise noted, to facilitate comparison with analysis. There are N identical stations.

For each simulation, to allow transients to die down, no statistics are collected for the initial 1 s. Thereafter, the simulation is run for 10 consecutive sub-runs, each of duration 1 s. This yields 95% confidence intervals on the order of 0.5% for aggregate statistics. In cases with large initial retransmission delays, longer run times are used.

We have seen that Ethernet performance is dependant on the parameter $a = \tau_p/T_p$, where T_p is the packet transmission time. Performance is high at low a and degrades as a increases. We use a packet length of $P = 40$ bytes, resulting in $a = 0.313$, to stress the network to its limits. Smaller values of a, corresponding to normal operating conditions, are also studied. Fixed size packets are used throughout. To simplify calculations, packet overhead is included in net throughput. Each station has B packet buffers. Inter-packet arrival times are exponentially distributed with mean θ. Packets arriving when the buffer is full are discarded. This is the non-feedback mode of operation described in Section 2.2.2.1. Network interface unit parameters are: carrier and collision detection time, $t_{cd} = 1\ \mu$s; inter-frame gap (minimum time between the end of one transmission and the start of the next), $t_{gap} = 1\ \mu$s; length of the jam signal on detection of a collision, $t_{jam} = 5\ \mu$s.

To describe the behaviour of throughput as offered load varies, we note that at any time a station is in one of three states: the *idle* state, while awaiting the arrival of a new packet; the *active* state, while contending for the channel; and the *backlogged* state, while incurring a retransmission delay after a collision. Note that the active state includes time spent in carrier sensing as well as in transmission.

For a given number of stations, N, the truncated back-off algorithm ensures stability for any value of G by discarding the fraction of packets that suffer a predefined maximum number of collisions (16 in the case of the standard Ethernet). The variation of η as a function of G is shown in Figure 4-14, with parameter B, the number of packet buffers in each station. For any B, initially throughput is equal to offered load. Here packet arrivals are spaced sufficiently that buffers are usually available, collisions are infrequent and

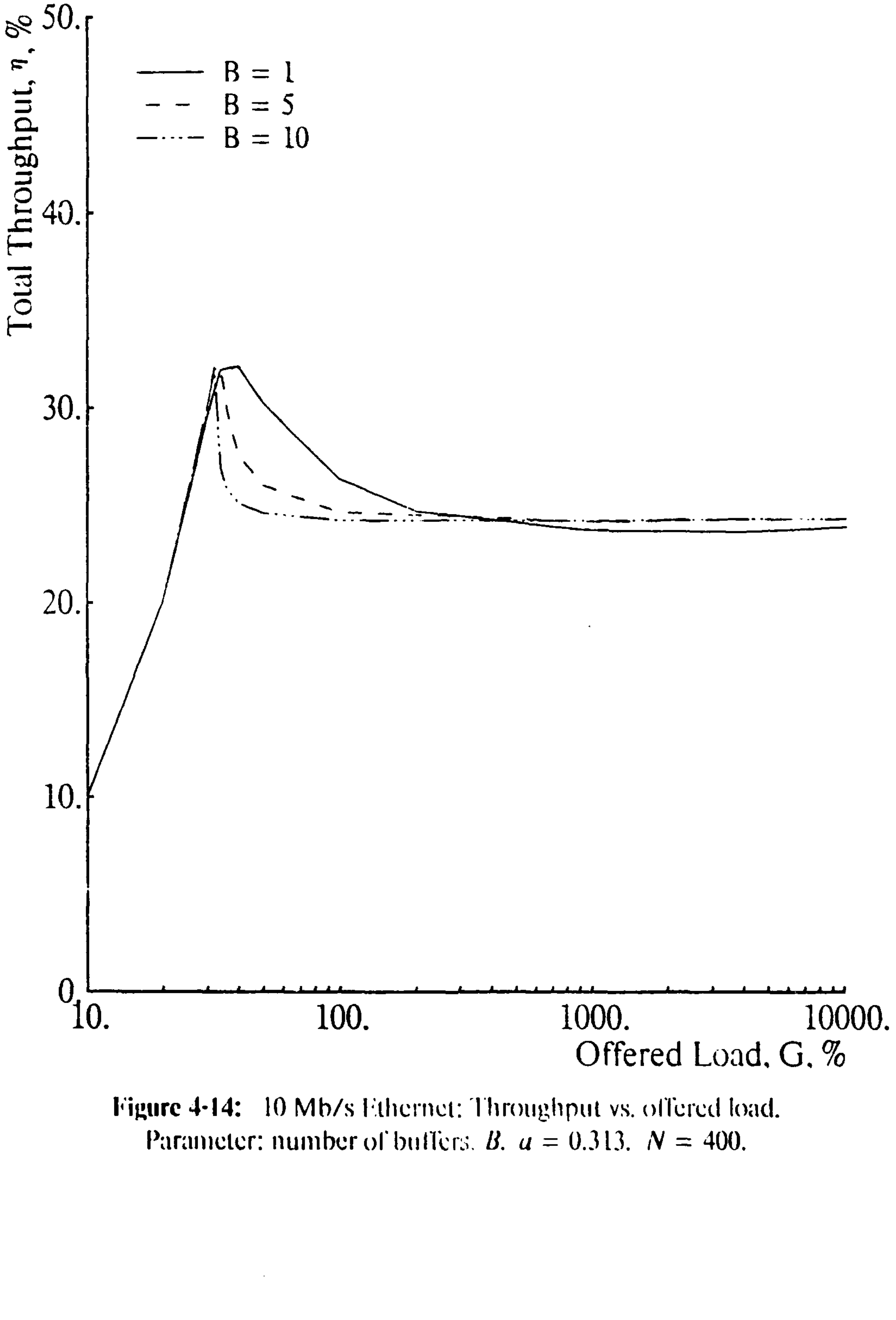


Figure 4-14: 10 Mb/s Ethernet: Throughput vs. offered load. Parameter: number of buffers, *B*. *a* = 0.313. *N* = 400.

delay is minimal. Stations alternate between the idle and active states with the holding time in the latter being the packet transmission time. Increasing the number of buffers has little effect because even with one buffer packets arrivals rarely occur when the buffer is occupied.

Further increase in G leads to a decrease in the mean station idle time as the time before the next arrival after a packet transmission decreases. The increased arrival rate causes an increase in the rate at which collisions occur. Owing to the binary exponential back-off algorithm, some stations suffer multiple collisions and spend greatly increased times in the backlogged state while other stations are successful without collisions. This effectively reduces the number of stations contending for the channel. Thus, throughput continues to increase to a peak.

As mean time between packet arrivals, θ, continues to decrease, the reduced idle time more than offsets the increased backlogged time and throughput drops to a stable value at heavy load. This stability at high values of G occurs because the mean time spent in the active and backlogged states is now much greater than the idle time. Thus, increasing G, and consequently reducing the idle time, has little effect. Likewise, at high loads with B = 1 the mean time in the idle state is close to zero and throughput is equal to the stable saturation value. Increasing B has little effect. The actual throughput depends on the ratio of the number of active stations to the number of backlogged stations and is determined by the back-off algorithm.

Increasing the number of buffers at moderate loads, i.e., in the region of the hump, has the effect of compressing the curve for B = 1 in Figure 4-14 horizontally. As more buffers are available, packets arriving while a station is active or backlogged enter the buffer rather than being lost. Thus the idle time is decreased, similar in effect to decreasing θ without increasing B. The peak value is approximately independent of B. It should be noted that the precise nature of this behaviour is a function of the protocol and in particular of the back-off algorithm. In measurements on a 3 Mb/s Ethernet with various values of a and 1 buffer, a hump was observed (Figure 4-1). In a study of multi-hop packet radio networks using CSMA, similar behaviour was observed [Shur 86].

The Ethernet back-off algorithm attempts to dynamically estimate the number of contending stations. For each packet, the estimate starts with 1 and is doubled with each successive collision. If the number of stations is fixed, theoretical studies show that stable and high performance is achieved for some fixed value of retransmission delay [Tobagi & Kleinrock 77, Tobagi & Hunt 80]. This suggests that an improvement on this algorithm is to use a higher initial estimate which can be derived from the collision history of previous packets. We assume that some such algorithm exists such that the initial estimate is 2^m, $m>0$. After the nth collision, the back-off is $\chi \times 51.2$ μs, where χ is a uniformly distributed random variable in the range $[0, M)$ such that:

$$M = \begin{cases} 2^{\min[\max(n,m),10]}, & \text{if } m < 10 \\ 2^{m}, & \text{if } m \geq 10 \end{cases}$$

For $m \geq 10$, we use M greater than the maximum of 2^{10} specified in the standard Ethernet because the latter value is found to be too small for our purposes as will be shown below.[12]

In Figure 4-15 throughput is plotted as a function of χ_{av} for a = 0.313 and N = 400. Throughput is seen to increase with χ_{av} initially and then to decrease. For a given set of parameters, with the system in equilibrium there is some combined mean arrival rate of new and collided packets. This determines the probability that an arrival will occur during the vulnerable period of a packet, causing a collision. As χ_{av} increases, stations spend longer periods in the backlogged state. Thus, the combined arrival rate decreases leading to a lower probability of collision and hence an increase in η. For sufficiently large values of χ_{av}, for some fraction of time all stations are backlogged and the network is idle resulting in a decrease in η. The optimum value of the mean initial back-off increases with N, being 194, 394 and 1024 for N = 200, 400 and 1000 respectively for the parameters in Figure 4-15. Note that these optimum values depend on parameters such as a and G.

Throughput at high loads in the standard Ethernet is a function of N. This dependence is shown in Figure 4-16. Throughput at saturation (G = 3600%) is plotted as

[12] We note that the Intel 82586 Ethernet controller chip allows M to be specified as $2^{min[n+m, 10]}$, for $m>0$. For $m \leq 10$, this uses the same initial value of χ as our algorithm. With multiple collisions, χ increases more rapidly but reaches the same maximum of 2^{10}.

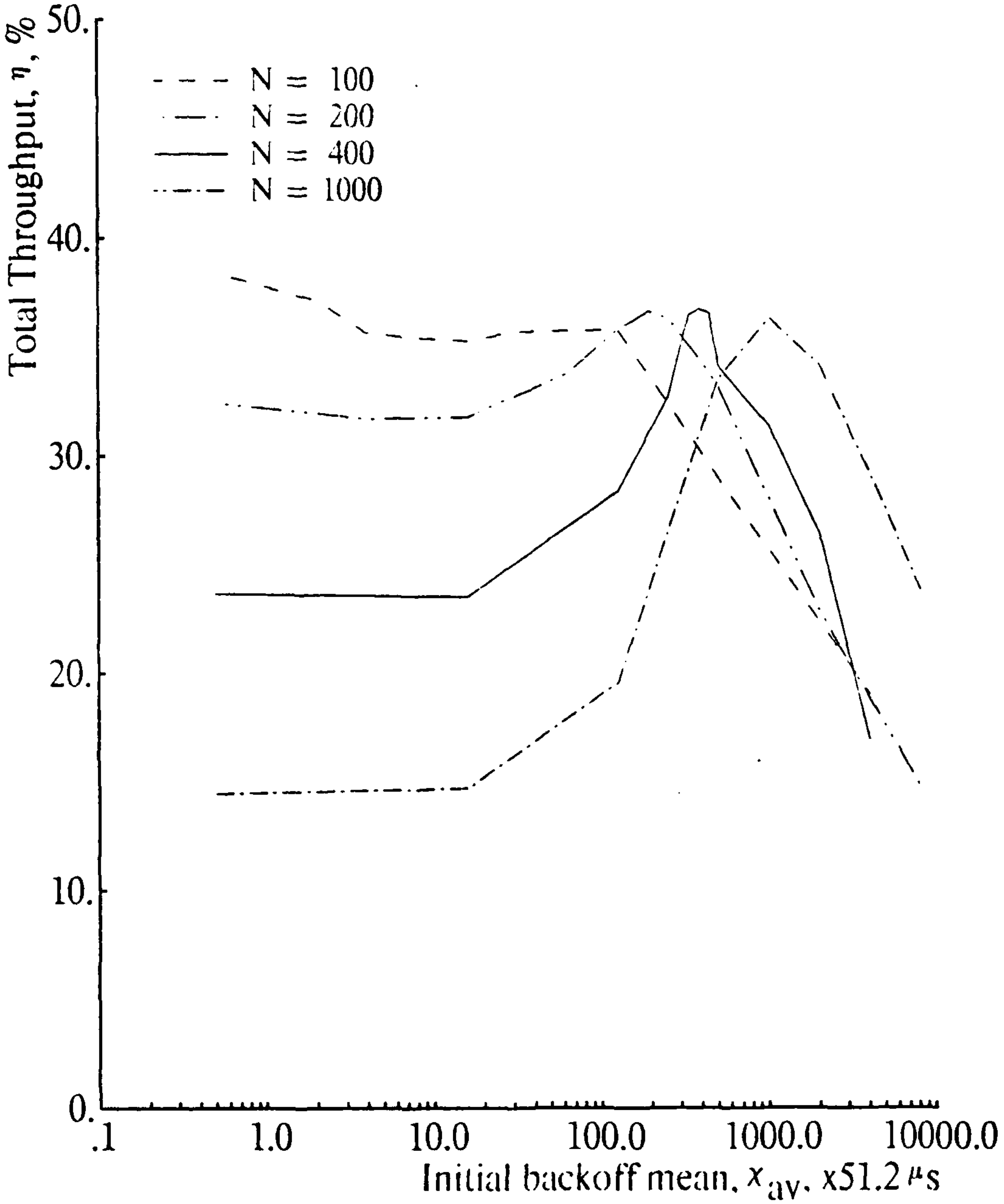


Figure 4-15: 10 Mb/s Ethernet: Throughput vs. Initial back-off mean. $a = 0.313$. $N = 400$.

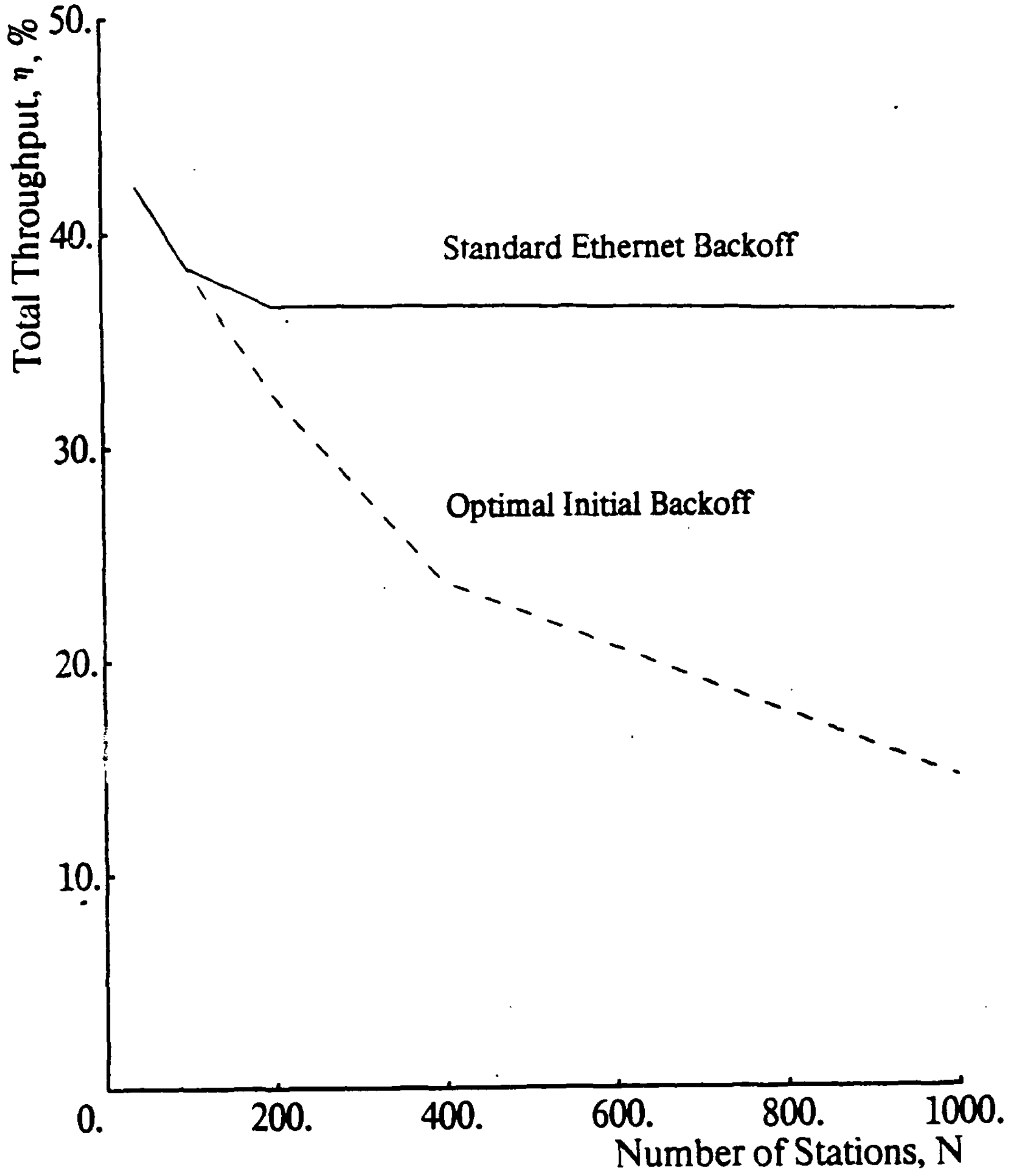


Figure 4-16: 10 Mb/s Ethernet: Throughput vs *N*
$a = 0.313$. Standard and modified back-off.

a function of N for the standard Ethernet and with the modified back-off algorithm. For the latter the optimum value of m is chosen for each N. By using the optimum back-off throughput remains approximately constant for N ranging from 200 to 1000. For small N, less than about 100, the standard Ethernet algorithm is optimal. For smaller values of a we expect that similar behaviour will be observed except that the value of N at which the standard Ethernet algorithm becomes sub-optimal will increase due to the relatively shorter vulnerable periods.

If a system is to support a large number of simultaneously active stations each generating a small load, for optimum throughput, a large initial value of χ_{av} should be chosen. For example, with $N = 1000$, the optimum is 1024. This yields a total throughput of 36% for $a = 0.313$ (Figure 4-15) and throughput per station, $\eta_{av} = 0.036\%$. If at some time fewer stations are active, the total throughput decreases but the throughput per station increases. In our example, with $N = 1000$, 400 and 200, $\eta_{av} = 0.036\%$, 0.078% and 0.14% respectively with average delays of 87, 40 and 25 ms respectively. Thus, individual stations are not adversely affected by the modified algorithm at light loads.

Comparison Revisited

The packet arrival processes used in the simulation and in the two analytical models are similar in that Poisson assumptions are made. However, retransmissions are handled differently as noted above. Hence it is not meaningful to compare performance at any specific offered load. Rather, we compare the maximum throughput predictions. The simulations were run with $G = 2000\%$, well within the saturation region. Table 4-7 shows throughput for the balanced star with $d = 2000$ m and 39 m. These correspond to $a = 0.313$ to 0.006 respectively. t_{jam} is constant at 5.0 μs. For the simulations, for $N = 400$, maximum throughput is shown for the standard and modified back-off algorithms described in the previous Section.

At small a, the analysis of Tobagi & Hunt corresponds well with the simulation. The analysis should be compared to the simulation of the modified algorithm at large N. The analytical prediction of 43.3% is higher than η_{max} of 36.6% with $N = 400$. The difference

	Simulation ($N=400$)		Analysis ($N=\infty$)	
a	Std back-off	Mod back-off	Tobagi & Hunt	Lam
0.313	23.7	36.6	43.3	28.2
0.006	61.9	73.9	70.0	63.3

Table 4-7: 10 Mb/s Ethernet: Simulation and Analysis, η_{max}. Balanced star topology. P = 40 bytes.

is due primarily to the assumption of slotted operation in the analysis.[13] If we consider the standard Ethernet, the analysis is seen to greatly overestimate η_{max} at $N = 400$. As N is decreased, the η_{max} increases and at $N = 40$ approximately matches that of the analysis. As N is further decreased, the analysis underestimates the standard Ethernet performance.

Considering Lam's analysis, we compare its prediction to η_{max} of the optimized algorithm with $N = 400$, as above. For the $a = 0.313$ and 0.006, η_{max} is consistently underestimated by 14 and 23% respectively. This is due in part to the pessimistic assumption that the slot duration is $t_{jam}+2\times\tau_p$. The use of a large slot size also leads to lower predictions than the model of Tobagi & Hunt. Considering the standard Ethernet, at large N, the analytical prediction is optimistic, at small N, pessimistic. The cross-over point occurs at about 300 for $a = 0.006$ and 0.313.

Balanced Star Topology

In the balanced star topology (Figure 4-13) the propagation delay between every pair of stations is τ_p. Hence the vulnerable period of every packet is $2\tau_p$ whereas in the linear bus topology $2\tau_p$ is an upper bound on the vulnerable period. Thus, other factors being the same, in the star topology stations would experience higher collision rates than in the linear bus topology and hence would achieve lower throughput, regardless of the distribution of stations on the linear bus network. In Table 4-8 η is shown for the two

[13] Considering an analysis of slotted and unslotted 1-persistent CSMA [Kleinrock & Tobagi 75], at a = 0.006, the slotted model leads to a 0.1% overestimation of η_{max} compared to the unslotted model; at a = 0.1, the difference increases to 4%, and at a = 0.3, to 15%.

topologies, with uniform distribution of stations in the case of the linear bus, and for several values of a. G is 2000%.

Network Parameters		Simulation	
d, m	a	Linear Bus*	Star
2000	0.313	61.1	42.2
640	0.100	67.7	60.1
64	0.010	71.6	70.7
39	0.006	71.7	71.3

*stations uniformly distributed.

Table 4-8: 10 Mb/s Ethernet: Star and linear bus topologies, η
N = 40 stations, P = 40 bytes, G = 2000%

For all values of a, throughput is lower in the case of the star. For large values of a, collisions due to the relatively large value of τ_p compared to T_p are the dominant cause of poor utilization. Here, the difference in throughput between the two topologies is appreciable, 8% and 20% for $a = 0.1$ and 0.313 respectively. For smaller values of a, the jam time of 5 μs becomes the dominant cause of poor utilization with the differences in the vulnerable period being small compared to T_p. Hence η is similar in the two topologies, though marginally lower in the case of the star.

4.3.4. Discussion

We have shown that the performance of the standard Ethernet algorithm in terms of maximum throughput is very poor at large N, compared to the analytic predictions of optimum throughput. This is due to the non-optimum rescheduling algorithm which always begins, independently for each packet, with a low value for the average rescheduling delay. A modified algorithm was presented, which results in significantly improved throughput. With these modifications, close correspondence is obtained between simulations of the Ethernet with large N and the predictions of two infinite population analytical models of CSMA/CD performance from the literature. Some residual discrepancies may be attributed to the use of slotting in the analyses and the effects of factors such as interface delays that are not explicitly considered in the analyses. This leads us to surmise that the modified algorithm is near optimal.

When compared to the predictions of these infinite population analyses, with large N, the standard Ethernet achieves lower performance, with small N, performance is higher, and at some intermediate value there is correspondence. The correspondence is better at small values of a. Thus, the analytical models may be used when rough estimates at low cost are desired and to provide insights into the protocol. Considerable care must be exercised in such use of the analytical models for large a. This is reinforced by the inconsistent accuracy of the analytical models when compared to our measurements. If accuracy is a concern, recourse must be had to the more expensive options of simulation or experimental measurement.

4.4. Station Locations

Due to the finite propagation delay of the signal on the network, two or more stations may sense the channel idle and start to transmit simultaneously, causing a collision. The probability of such interference is dependant on the distances between stations. We use simulation to investigate this dependence further. It is shown that unbalanced distributions can lead to significant differences in performance achieved by individual stations.

4.4.1. The Configurations

The simulation parameters are chosen to resemble the 10 Mb/s Ethernet and are described in Section 4.3.3, page 66. We use fixed size packets of length $P = 40$ bytes, resulting in $a = 0.313$, to stress the network to its limits. All stations are identical, except for location. The N stations are numbered from 1 to N. For ease of notation and description we assume, without loss of generality, that the station numbers increase monotonically from one end of the network, referred to as the left end, to the other or right end. The stations are distributed along the network in the following configurations:

- *Uniform:* stations are spaced uniformly along the length of the network, with spacing between every pair of adjacent stations being d/N m (Figure 4-17).
- *Equal-sized clusters:* the stations are divided into n_c clusters, with each cluster having N/n_c stations. The centres of the clusters are uniformly spaced along the network. The stations in a cluster are spaced uniformly, with adjacent stations being less than or equal to d/N m apart (Figure 4-18). The *uniform* configuration above is a special case with cluster size equal to 1.
- *Unequal-sized clusters:* as above except that the number of stations in each cluster is not the same (Figure 4-19).

For each simulation, to allow transients to die down, no statistics are collected for the initial 1 s. Thereafter, the simulation is run for 10 consecutive sub-runs, each of duration 1 s. This yields 95% confidence intervals on the order of 0.5% for aggregate statistics.

4.4.2. Simulation Results

We now examine the performance of the configurations described above. First we consider various numbers of equal-sized clusters, then we examine the effects of the intra-cluster spacing, and finally we consider clusters of differing sizes. We end with a comparison of the star and linear bus topologies.

It is useful to define the *vulnerable period* of station i, the period during which another station would have to start transmission in order to cause a collision with i's transmission. Assume that the propagation time from station i to the furthest end of the network is τ and that station i starts transmission at time t. A packet from station j situated at the furthest end of the network will collide with i's transmission only if j starts transmission in the interval $[t-\tau, t+\tau]$. This interval is the vulnerable period of i. The probability that j causes a collision with i's packet is simply the probability that j starts transmission during i's vulnerable period. Note that a station closer to i than the furthest end of the network would have a shorter interval about t during which it could cause a collision with i's packet.

The number of stations, N, is 40. At light loads, i.e., G less than network capacity,

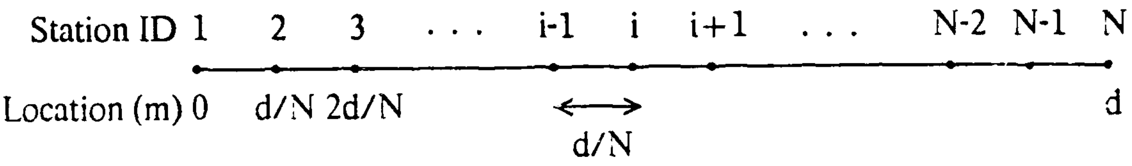


Figure 4-17: Station Distribution: Uniform Spacing

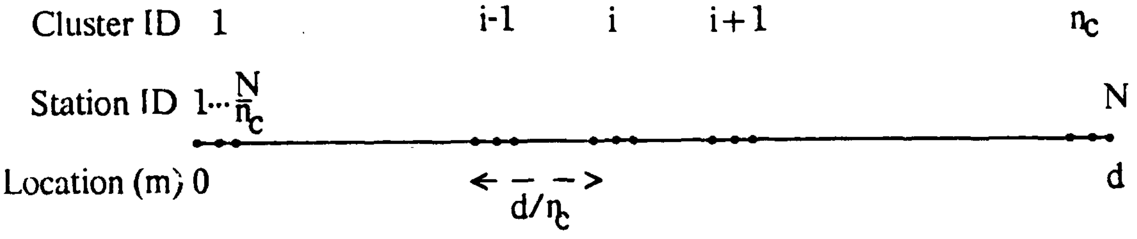
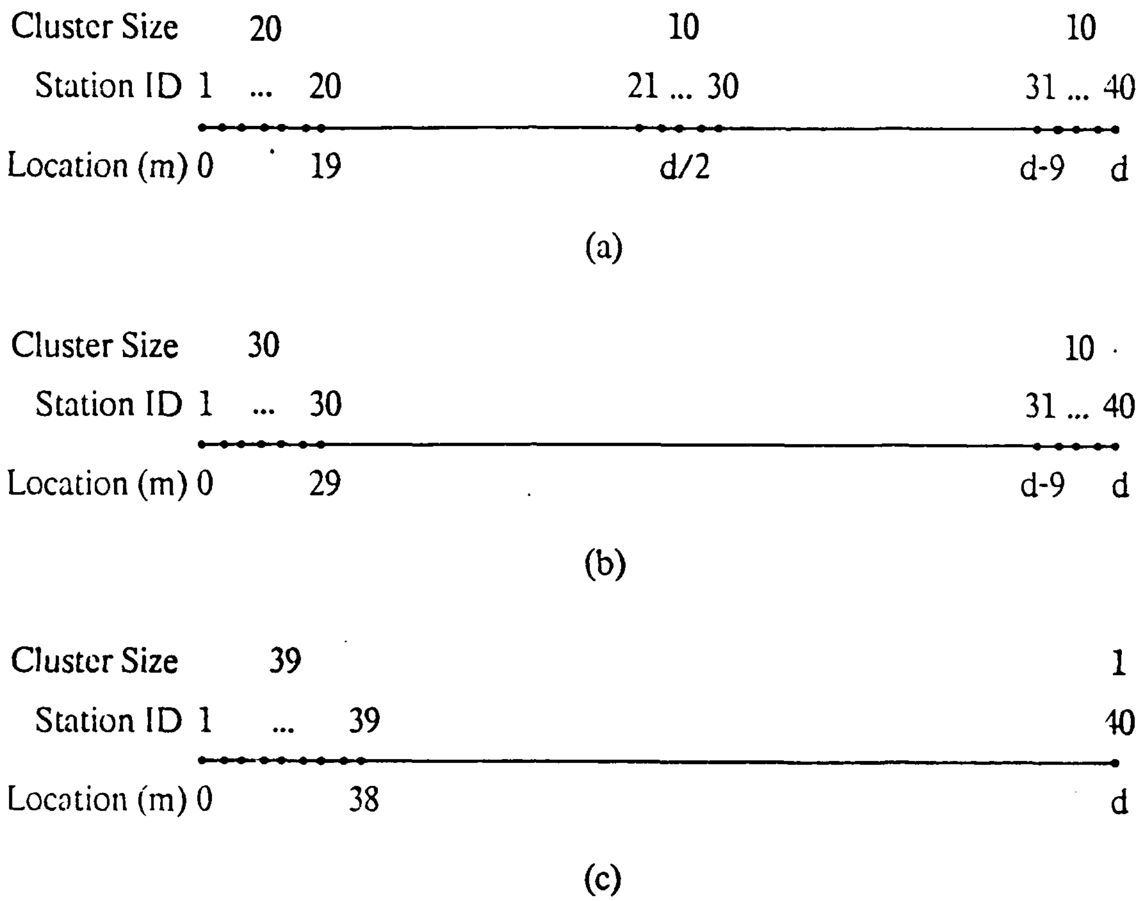


Figure 4-18: Station Distribution: Equal-sized Clusters



(a)

(b)

(c)

Figure 4-19: Station Distribution: Unequal-sized Clusters

measurements (Section 4.2) and simulations (Section 4.3) have shown that delay is minimal and the throughput achieved by station i is approximately G_i. To bring out performance differences, we consider moderately heavy loads. The performance effects to be discussed were noted in simulations with G ranging from 100% to 2000%, though with differing magnitudes. Further, total throughput is found to reach a saturation value as G exceeds 100%, without changing substantially as G is increased to greater than 2000%. Hence, in the simulations, θ is chosen to yield a total offered load, G, of 400% of network capacity, well within the saturation region.

Equal-sized Clusters

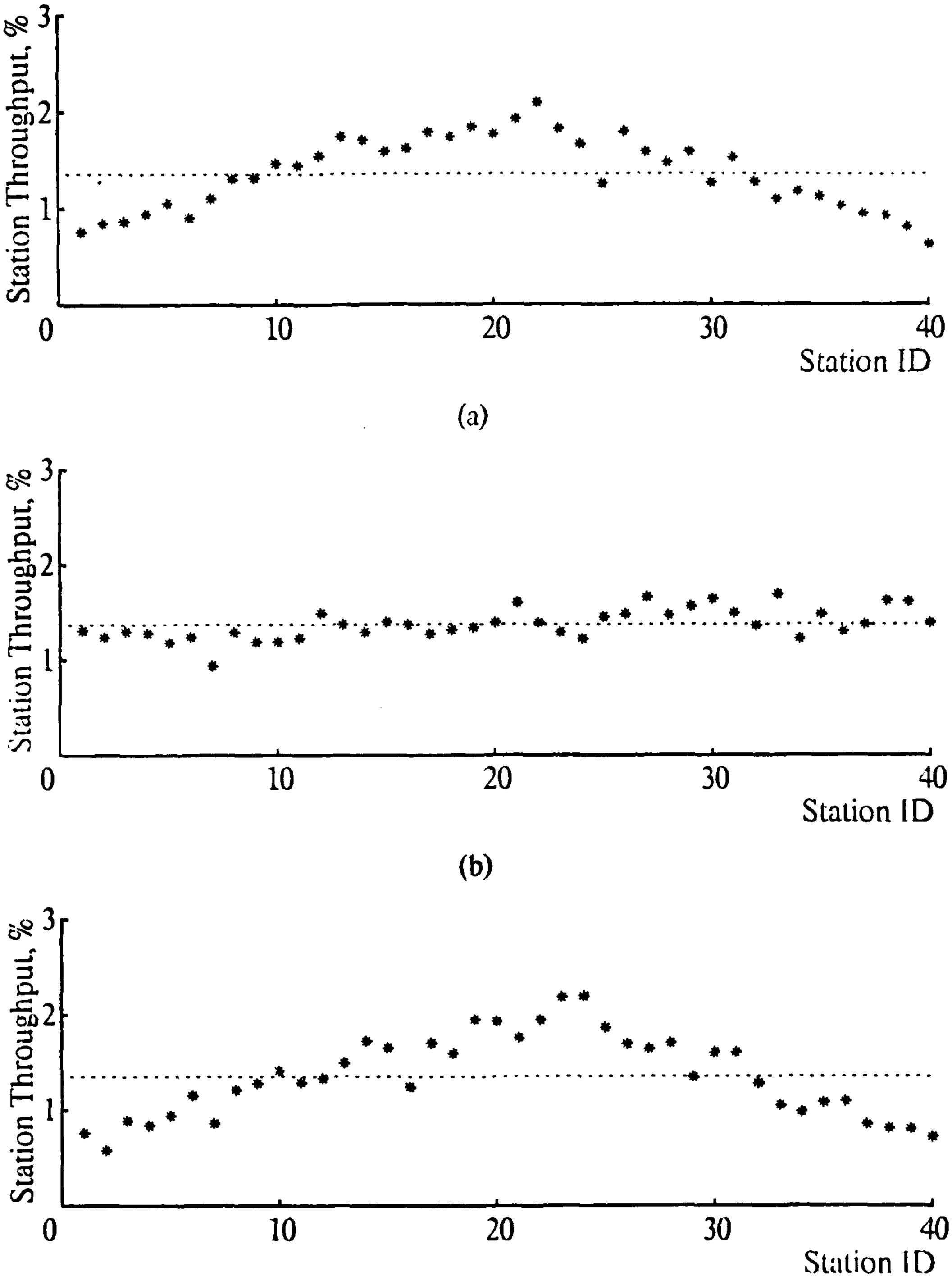
Stations are clustered into 2 to 10 equal-sized clusters with an intra-cluster spacing of 1 m. The clusters are uniformly distributed on the network. Table 4-9 shows the total throughput, η_{total}, average throughput per station, $\eta_{av} = \eta_{total}/N$, and the average packet delay, D. Also shown are the standard deviations of the metrics per station.

Configuration	η_{total}, %	η_{av}, %		Delay, ms	
		Mean	Std dev	Mean	Std dev
Uniform, 2 km	54.6	1.36	0.38	1.69	0.62
2 equal clusters	54.9	1.37	0.16	1.52	0.21
3 equal clusters	52.3	1.31	0.47	1.76	0.61
5 equal clusters	53.0	1.32	0.45	1.73	0.66
10 equal clusters	54.0	1.35	0.43	1.66	0.65

Table 4-9: 10 Mb/s Ethernet: Stations in Equal-Sized Clusters
$N = 40$ stations, $G = 400\%$

It is seen that the number of clusters has minimal effect on mean throughput and delay. In the case of 2 clusters, the performance is marginally better than in the other cases. The standard deviation of the individual measures is also almost constant. Again, the 2-cluster configuration is an exception, with lower variation. In the 2 cluster configuration, every station has 19 other stations no more than 19 m away and 20 stations at distances of between 1962 and 2000 m. Since the distance of the latter group is two orders of magnitude greater than that of the former, to a first approximation the vulnerable periods of all stations are equal. Further, we may consider that collisions arise solely as a result of transmissions by the 20 distant stations. Hence, every station is statistically identical and variance of individual performance measures is expected to be low. This equivalence does not occur with n_c greater than 2.

Figure 4-20 shows the throughput measured by each station plotted against the station number for various cluster sizes. The average throughput per station is shown by a dotted line. Likewise, individual delays are plotted in Figure 4-21. (Note that the abscissa is not distance.) In all cases, the stations at the centre of the network obtain the highest throughput with lowest delay, while the stations at the ends obtain a lower share of the total throughput. This occurs because the vulnerable period ranges from τ_p for a station at the centre to $2\tau_p$ for a station at the end. Thus, centrally located stations suffer fewer collisions per packet and hence obtain higher throughput, while stations at the ends suffer more collisions and hence experience higher delays due to retransmissions and longer back-offs.

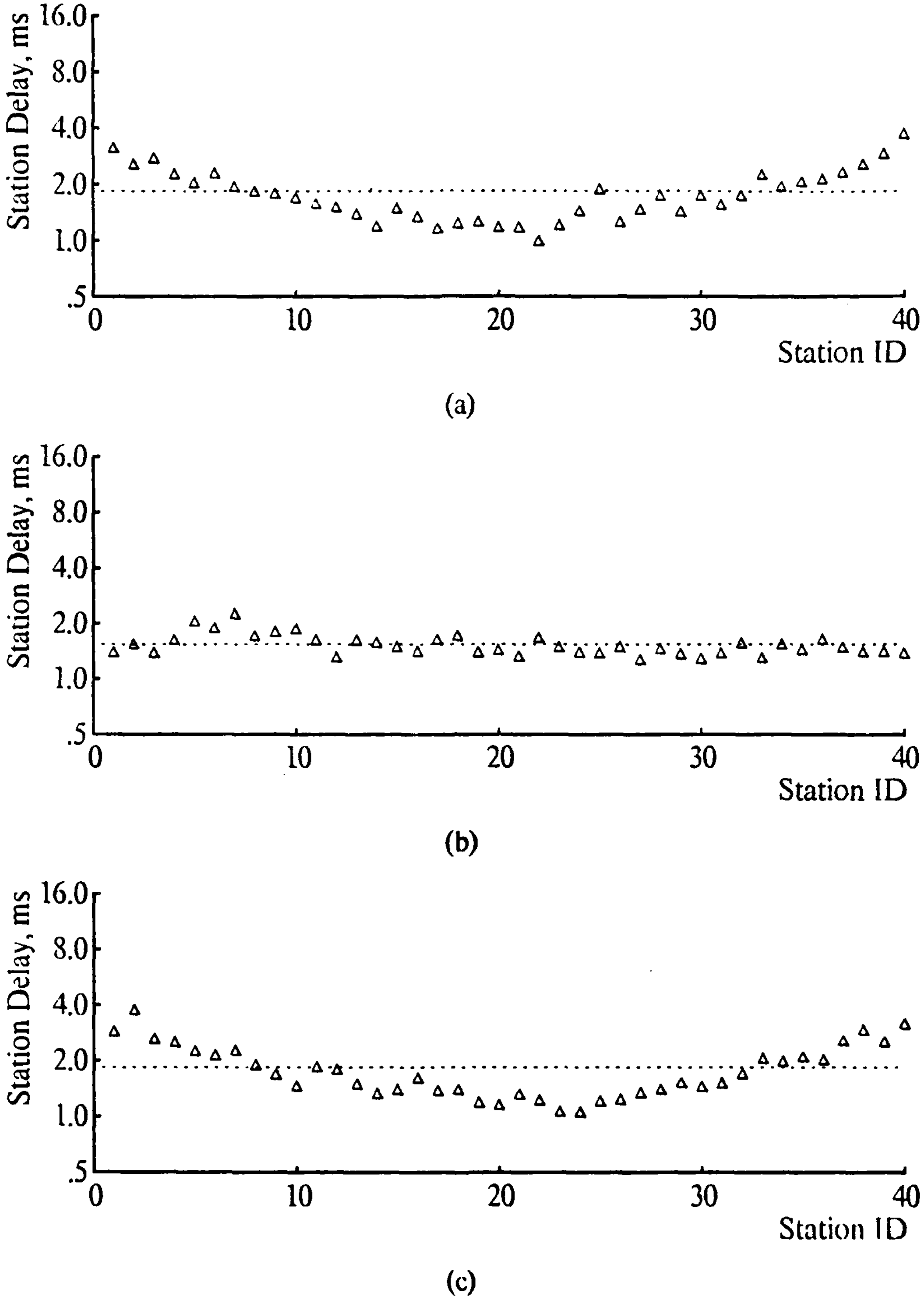


(a)

(b)

(c)

Figure 4-20: 10 Mb/s Ethernet: Individual Throughputs, Equal-Sized Clusters
N = 40 stations, G = 400%.
(a) Uniform. (b) 2 Clusters. (c) 10 Clusters.



(a)

(b)

(c)

Figure 4-21: 10 Mb/s Ethernet: Individual Delays, Equal-Sized Clusters
$N = 40$ stations, $G = 400\%$.
(a) Uniform. (b) 2 Clusters. (c) 10 Clusters.

As discussed above, due to the disparity between intra- and inter-cluster distances, the stations within a single cluster obtain similar performance. Comparing the 10 cluster and the uniform configurations, it is seen that the performance of a station in a cluster located at a distance of x m from the left end of the network is similar to that of the station at the same location in the uniform case. This is noted in the other clusterings not shown here.

Intra-Cluster Spacing

We examine the effects of varying the intra-cluster spacing with 5 clusters of 8 stations each. The clusters are uniformly spaced along the 2 km network. The intra-cluster spacing is set to 1, 5, 20 and 50 m. (Note that a 50 m spacing corresponds to a uniform distribution of the stations on the network). Table 4-10 shows that there is little difference between aggregate statistics. At the maximum spacing of 50 m, performance is marginally improved. The same is true of individual measures (not shown).

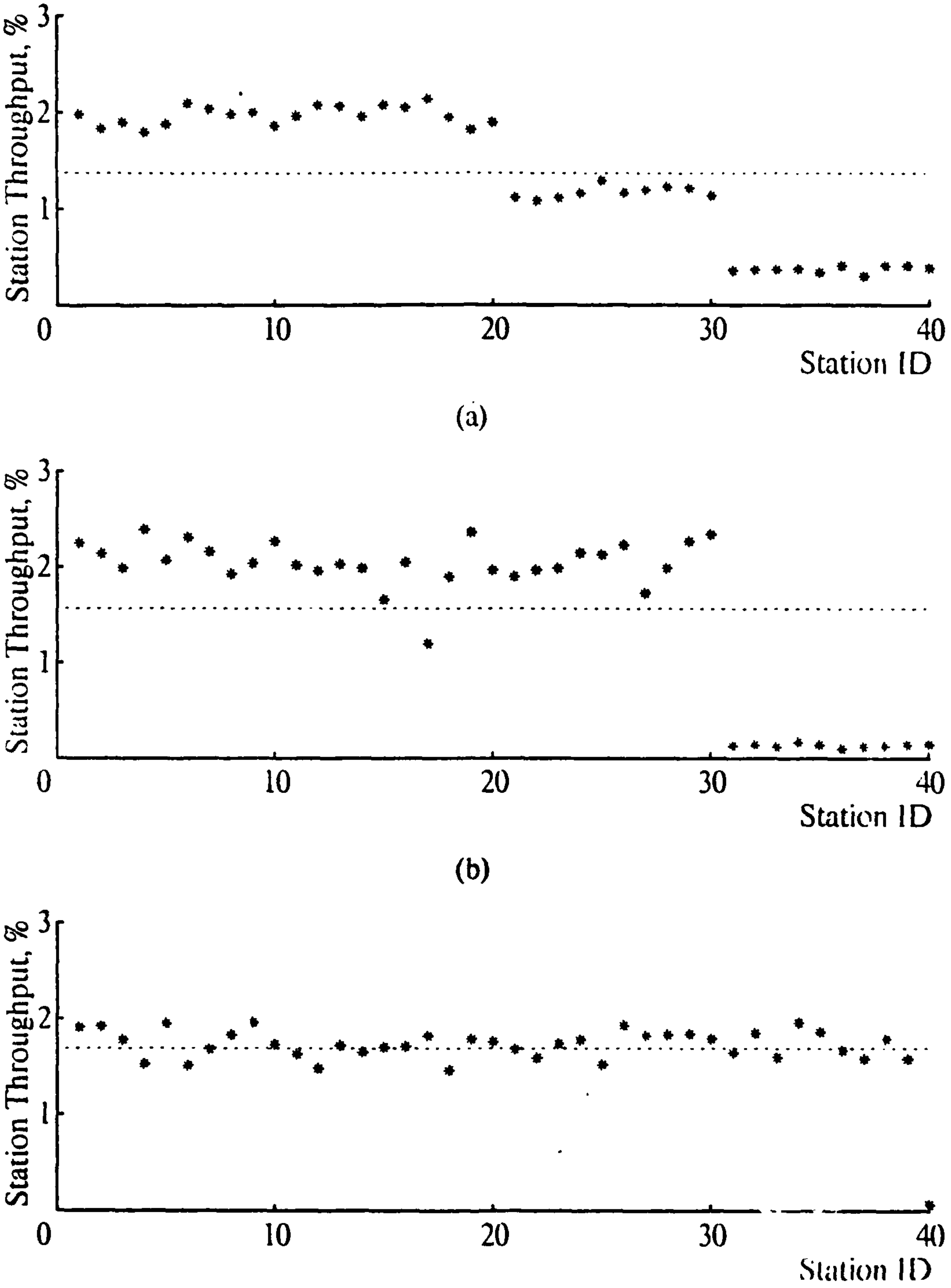
Configuration	η_{total}, %	η_{av}, %		Delay, ms	
		Mean	Std dev	Mean	Std dev
1 m spacing	53.0	1.32	0.45	1.73	0.66
5 m spacing	53.2	1.33	0.42	1.75	0.64
20 m spacing	53.3	1.33	0.39	1.71	0.53
50 m spacing	54.6	1.36	0.38	1.69	0.62

Table 4-10: 10 Mb/s Ethernet: 5 equal clusters, various intra-cluster spacings
$N = 40$ stations, $G = 400\%$

Unequal-sized Clusters

The configurations considered so far have been symmetrical about the centre of the network. We now consider asymmetrical distributions of stations. We start with 3 equally spaced clusters of equal sizes and then progressively move stations from the right to the left of the network resulting in the distributions shown in Figure 4-19. In the most asymmetrical configuration considered, we have all 40 stations in the left-most 39 m of the network (effectively, a 39 m network with uniformly spaced stations).

The mean and standard deviation of throughput and delay are shown in Table 4-11,

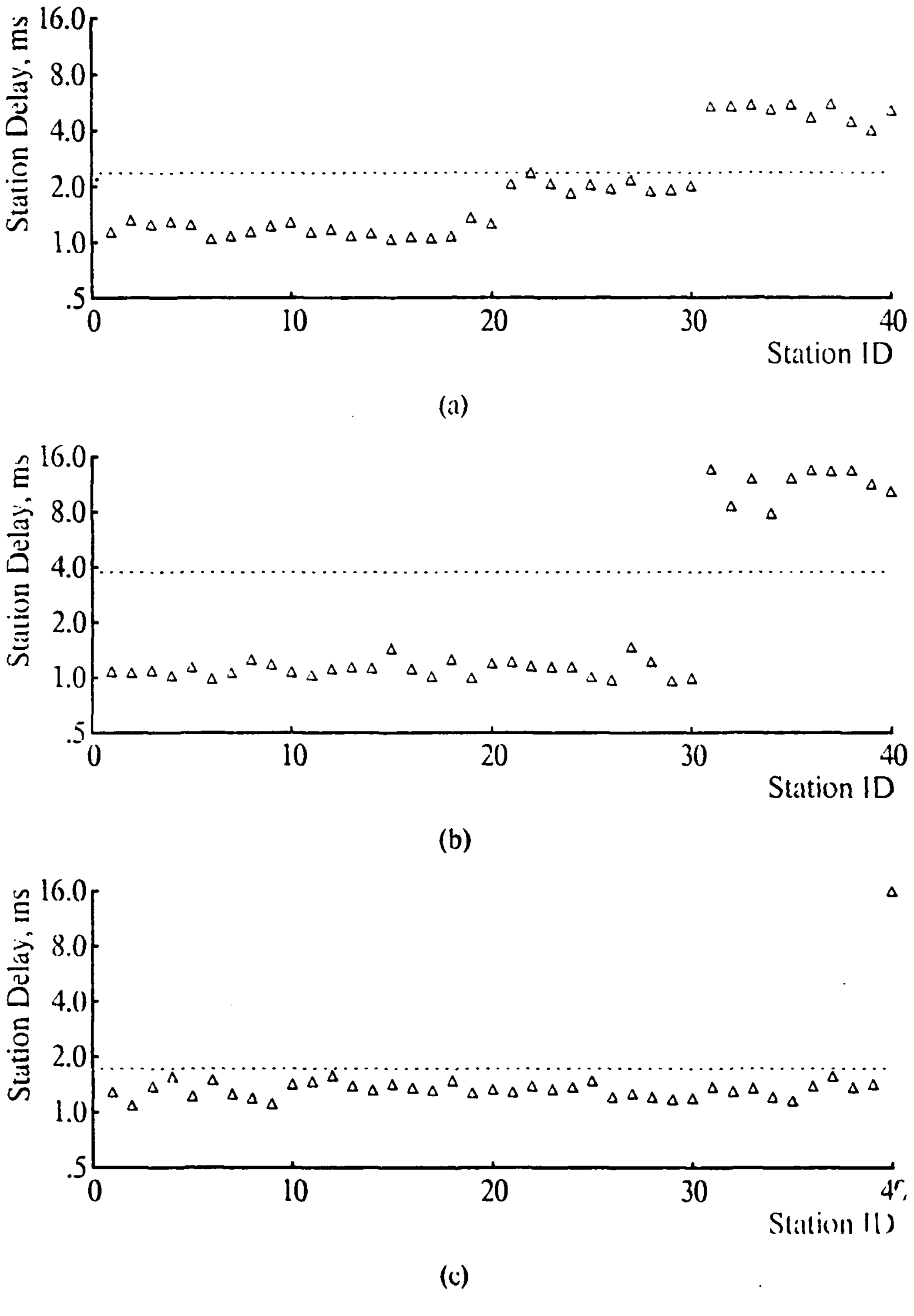


(a)

(b)

(c)

Figure 4-22: 10 Mb/s Ethernet: Station Throughputs, Unequal-Sized Clusters
N = 40 stations, G = 400%. Cluster sizes:
(a) 20 + 10 + 10 Stations. (b) 30 + 10 Stations. (c) 39 + 1 Stations



(a)

(b)

(c)

Figure 4-23: 10 Mb/s Ethernet: Individual Delays, Unequal-Sized Clusters
$N = 40$ stations. $G = 400\%$. Cluster sizes:
(a) 20 + 10 + 10 Stations. (b) 30 + 10 Stations. (c) 39 + 1 Stations

while the distributions of individual measures are plotted in Figures 4-22 and 4-23. As the asymmetry increases, the net throughput increases markedly from 52% in the fully symmetrical case to 68% in the case when all the stations are at one end. Considering the performance distributions, stations in the larger clusters obtain a disproportionately large share of the net throughput. When a station in a cluster starts to transmit, the signal propagates in a short period to the ends of that cluster. Hence, the vulnerable period of the station to collision from other stations in the same cluster is small. The packet is vulnerable to collision from stations in other clusters for a relatively much longer time. Thus, a station in a large cluster is less vulnerable than one in a small cluster. In the 20+10+10 case, the two 10-station clusters obtain significantly different performance, with the one in the centre obtaining higher performance due both to being in the centre and to being closer to the large 20-station cluster.

Configuration	η_{total}, %	η_{av}, %		Delay, ms	
		Mean	Std dev	Mean	Std dev
3 equal clusters, 2 km	52.3	1.31	0.47	1.76	0.61
20 + 10 + 10	54.9	1.37	0.68	1.62	1.65
30 + 10	63.6	1.59	0.87	1.33	4.75
39 + 1	77.7	1.69	0.30	1.34	2.48
Uniform, 39 m	67.9	1.70	0.14	1.36	0.14

Table 4-11: 10 Mb/s Ethernet: Stations in Unequal-sized clusters
N = 40 stations, G = 400%

4.4.3. Discussion

We have examined the sensitivity of the Ethernet to various distributions of stations on the network. With stations uniformly distributed along a linear bus, stations at the centre obtain better performance than stations at the ends. Clustering the stations in equal-sized clusters has little effect on total throughput or on the individual throughput as a function of network location. Increasing the intra-cluster spacing also has little effect except to increase the variance in performance within each cluster. Introducing inequalities in the cluster sizes increases total throughput, with the stations in the larger clusters obtaining a greater than proportionate share at the expense of stations in the smaller clusters.

An implication of these results is that a configuration such as a cluster of workstations at one end of an Ethernet simultaneously accessing a server at the remote end may lead to unexpected congestion as the server experiences higher than normal collision rates. In such a situation it may be advantageous to alter the back-off strategy of the server's network interface unit to mitigate the effects of collisions. The extent to which these results hold in the case of heterogeneous stations is a matter for further investigation.

4.5. Voice Traffic: Measurement and Simulation

We now turn our attention to the performance of the Ethernet under real-time constraints. First, we show that an experimental 3 Mb/s Ethernet can support packetized voice transmission, achieving high throughput while satisfying real-time constraints. (Section 4.5.1). We then use simulation to extend this work to higher bandwidths with integrated voice/data traffic (Section 4.5.3). In Section 4.5.2, we investigate the effect of various values of P_{min}, the minimum voice packet length of our voice packetization protocol.

4.5.1. 3 Mb/s Ethernet: Voice Performance

We use measurements on a experimental 3 Mb/s Ethernet to show that the Ethernet protocol can handle voice traffic satisfactorily. The network used is described in Section 4.1.2, the experimental techniques are summarized in Sections 3.3 and 4.2.1. In this Section we summarize the findings that have been presented in greater detail elsewhere [Gonsalves 83].

During each experiment, each participating station generates samples at a constant rate, V b/s, emulating the output of a voice digitizer without silence suppression. For accurate timing, the coder emulator was written in microcode and can generate two 8-bit samples every $38.08n$ μs, for $n = 1, 2, \ldots$. Thus, V can be any sub-multiple of 420 kb/s. For example, $n = 4$ and 6 yield $V = 105$ and 70 kb/s respectively. Silence suppression is not modelled and the experiments are conducted in the absence of data traffic. For convenience, we chose $V = 105$ kb/s since this value enables us to generate a higher load with a given number of stations than would be possible with a lower value of V. Further, our experiments showed that performance results obtained with $V = 105$ kb/s scale linearly to $V = 70$ kb/s provided that packet delays are multiplied by a factor of $105/70 = 1.5$. Note that this is not a general result but is valid for the conditions under which the experiments were conducted.

For several values of the parameters D_{min} and D_{max} of the packet-voice protocol, we plot in Figure 4-24 loss as a function of the number of stations. In all cases, it is seen that there is no loss for small values of N_v. Then, there is a well-defined knee at which loss starts. Thereafter, loss increases rapidly with increase in N_v. The system voice capacity with a maximum acceptable loss of φ, $N_{v_{max}}^{(\varphi)}$, is defined to be the value of N_v at which loss equals φ. This value is dependant on other parameters such as D_{max}. Since loss on the order of 1% is acceptable to listeners (Section 2.2.1.2), operation in the region to the left of the knee on any curve is acceptable while operation to the right is undesirable. With a tight delay constraint of 5 ms, the knee is seen to occur at about 18 stations. This is lower than the theoretical maximum of $\lfloor 3.0/0.105 \rfloor = 28$ stations because the short packet length of 64 bytes results in an appreciable collision rate. With a more relaxed $D_{max} = 80$ ms, $N_{v_{max}}$ ranges between 24 and 27 stations for D_{min} between 5 and 40 ms. Here, utilization of the channel is close to the theoretical maximum.

By applying the conversion factor of 1.5 to scale our results to $V = 70$ kb/s, we conclude that with a delay constraint of 7.5 ms, 27 stations could be supported while with a constraint of 120 ms, about 36 stations could be supported. We will show later that the use of silence suppression can increase these numbers by a factor of about 2 (Section 4.5.3.1).

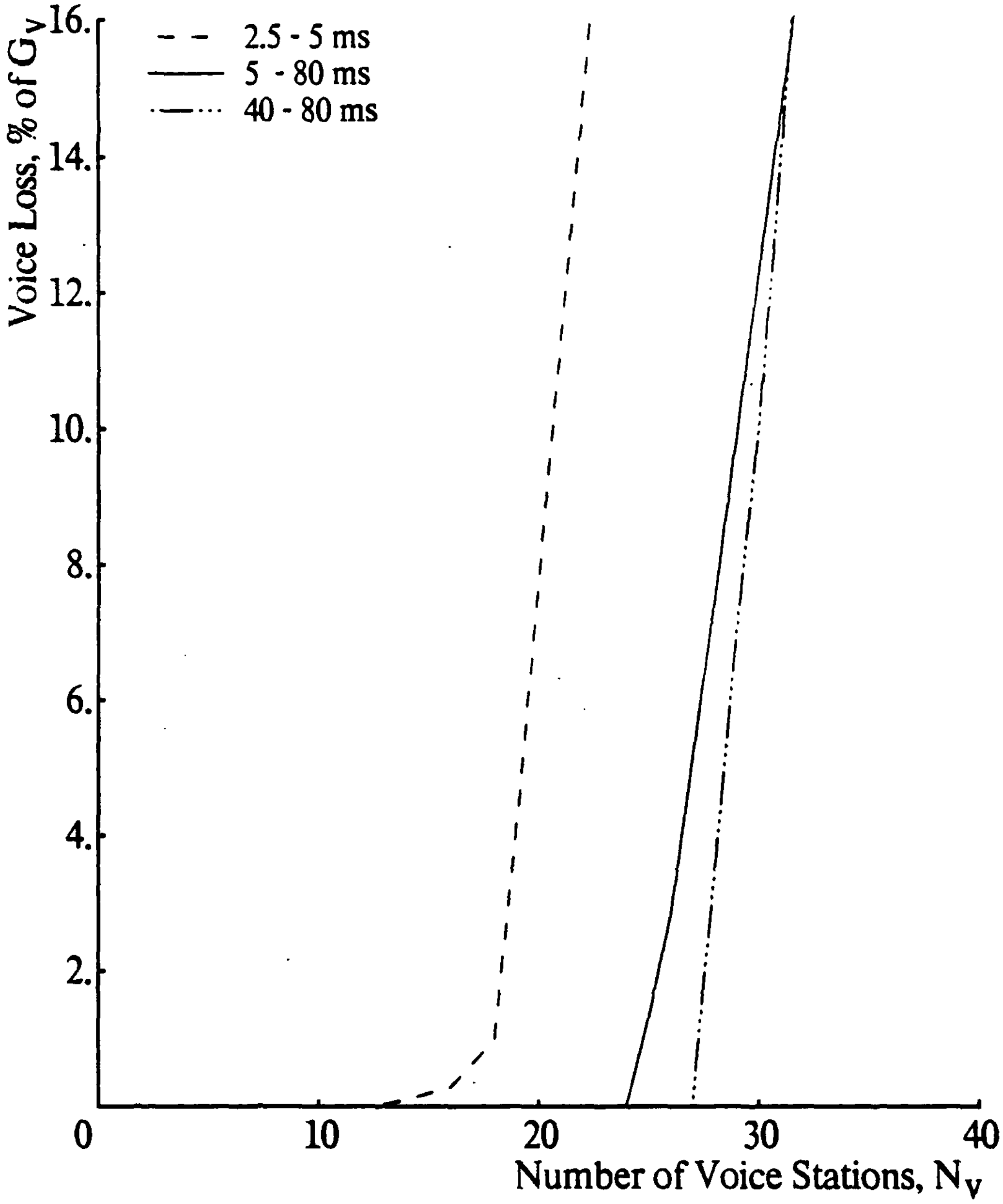


Figure 4-24: 3 Mb/s, 0.55 km Ethernet: Loss vs. N_v. Measurements. V = 105 kb/s. Parameters: D_{min}, D_{max}. Without silence suppression. G_d = 0%.

4.5.2. Minimum Voice Packet Length

The minimum voice packet length, P_{min}, is a network-dependant parameter of the voice packetization protocol (Section 2.2.1.3). In selecting an optimal value of P_{min}, it is desirable that $N^{(\varphi)}_{v_{max}}$ be maximized, that the average and maximum clip lengths be minimized and that the adverse impact on data traffic performance be minimized. We empirically determine an optimal value for P_{min} for each value of D_{max}. While the optimum is dependant on other factors such as whether or not silence suppression is used, we expect D_{max} to be the prime determinant of the optimum. First we discuss measurement results for the 3 Mb/s Ethernet under the conditions discussed in Section 4.5.1 and then extend this via simulation to the conditions to be used in the voice/data evaluation in Section 4.5.3.

In Table 4-12 we show the maximum number of voice stations with loss of 1 and 5% for several values of P_{min} obtained in measurements on a 3 Mb/s, 0.55 km Ethernet. D_{max} is 80 ms. V is 105 kb/s and silence suppression is not used. It is seen that both $N^{(1)}_{v_{max}}$ and $N^{(5)}_{v_{max}}$ increase with increase in P_{min} indicating that a large value of P_{min} should be chosen.

D_{min}	$\varphi =$	1%	5%
2.5 ms		25	27
10 ms		26	27
40 ms		27	28

Table 4-12: 3 Mb/s, 0.55 km Ethernet: Voice Capacity at φ = 1, 5%. Measurements. V = 105 kb/s. D_{max} = 80 ms. Parameter: D_{min}. Without silence suppression. G_d = 0%.)

Owing to the inflexibility of measurement, we now turn to simulation to investigate more thoroughly the effects of varying P_{min}. For a 10 Mb/s, 1 km Ethernet, we present simulation results under the following conditions: D_{max} = 20 ms, without silence suppression, G_d = 20%. From this we obtain a value for P_{min} that is used for all simulations with D_{max} = 20 ms in Section 4.5.3. Similar optimizations yield values of

P_{min} for $D_{max} = 2$ and 200 ms. Note that the minimum voice delay, D_{min}, is related to P_{min} by $P_{min} = VD_{min}$.

In Figure 4-25, $N_{v_{max}}^{(\varphi)}$ is plotted as a function of D_{min} for various values of φ. For a given value of φ, as D_{min} is increased from zero to D_{max}, $N_{v_{max}}$ increases to a peak and then decreases. When D_{min} is small, bandwidth wasted due to contention and packet overhead is large compared to the useful data in each packet. Even though the packet length increases with load due to the voice protocol for packets that suffer several collisions, some packets are successful after few collisions. Thus the mean voice packet length remains small. When D_{min} is close to D_{max}, any contention delay causes the packet length to exceed the maximum and loss occurs. This causes a decrease in $N_{v_{max}}$ that is larger for small values of φ. The value of D_{min} at which $N_{v_{max}}$ is maximum increases with φ.

From Table 4-13, we see that the mean and maximum clip lengths decrease with increase in D_{min}. Values at $\varphi = 1\%$ are shown in the Table. Other values of φ show a similar trend. While from this point of view it is desirable to choose D_{min} close to D_{max}, from the point of view of $N_{v_{max}}^{(\varphi)}$, a smaller value is preferred.[14] For a given φ, D_{min} has only marginal effect on data measures, η_d and D_d. Thus, for $D_{max} = 20$ ms, we choose the value of 16 ms as near optimal for D_{min}. Similarly, the near optimal values of D_{min} chosen for $D_{max} = 2$ and 200 ms are 0.75 and 160 ms respectively.

4.5.3. 10 Mb/s Ethernet: Voice/Data Performance

Having seen that the 3 Mb/s Ethernet is a viable alternative for packetized voice applications, we extend the study to higher bandwidths and consider integrated voice/data applications as described in Chapter 2. For the parameter ranges described in Section 2.4, we examine first voice traffic performance measures in Section 4.5.3.1 and then data traffic measures in Section 4.5.3.2.

[14] We consider small values of φ as these are more usual in voice telephony.

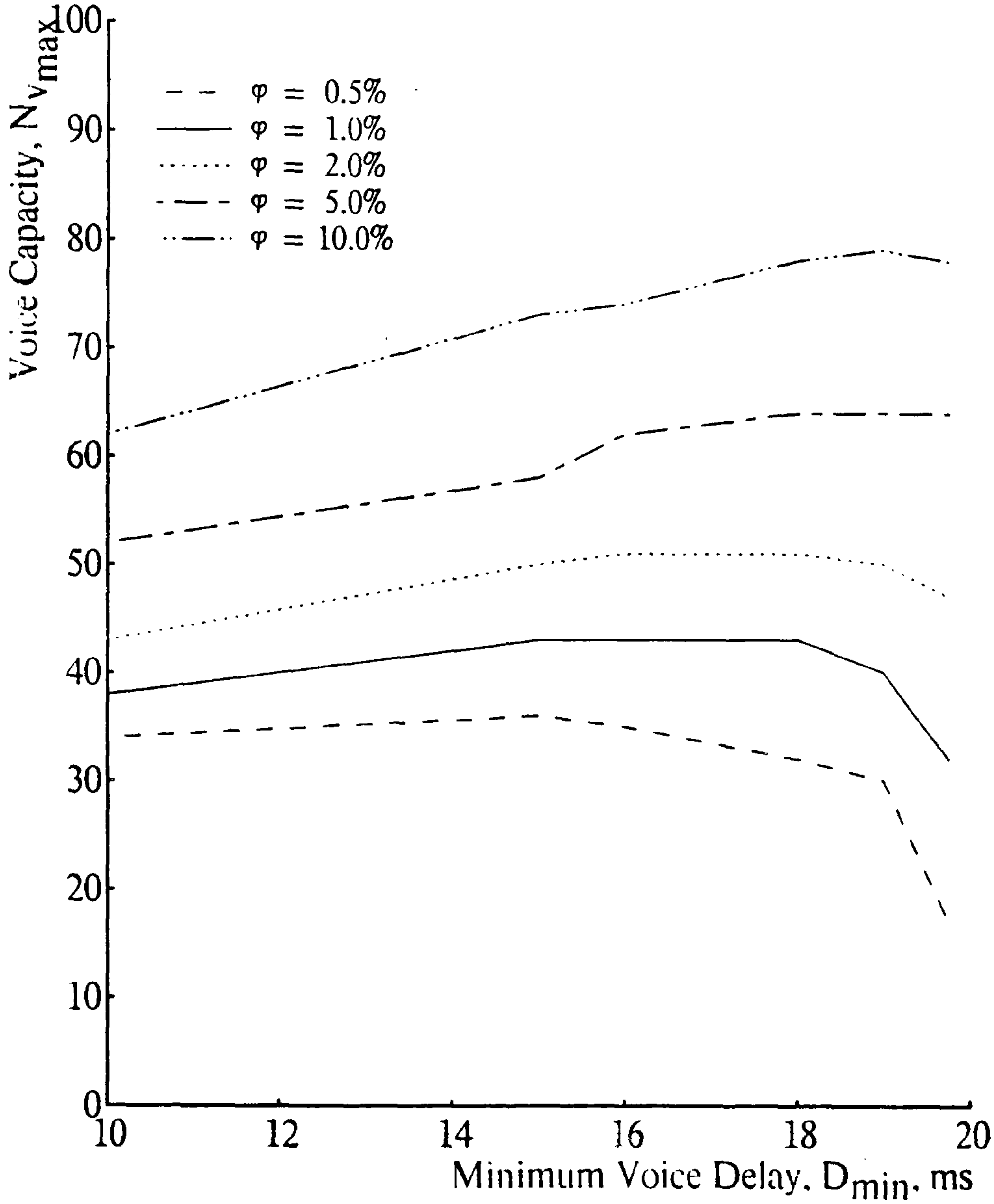


Figure 4-25: 10 Mb/s, 1 km Ethernet: $N_{v_{max}}$ vs D_{min}, parameter, φ. Without silence suppression. $D_{max} = 20$ ms. $G_d = 20\%$.

D_{min} (ms)	$N^{(1)}_{v_{max}}$	Clip Length, T_l (ms)		
		Mean	Std dev	Max
10.0	38	14.9	23.7	172
15.0	43	8.4	16.8	188
16.0	43	6.9	13.6	153
18.0	43	4.1	8.7	140
19.0	40	2.5	5.9	111
19.75	32	1.3	2.8	82

Table 4-13: 10 Mb/s, 1 km Ethernet: Clip lengths at $\varphi = 1\%$. Without silence suppression. $D_{max} = 20$ ms. $G_d = 20\%$. Parameter, D_{min}.

	D_{max} (ms)	$N^{(1)}_{v_{max}}$	Theor. Max. $\lfloor C/V \rfloor$
3 Mb/s, 0.55 km	8.2	29	46
Measurement*	131	44	
10 Mb/s, 1 km	20	119	156
Simulation	200	152	
100 Mb/s, 5 km	20	130	1562
Simulation	200	1100	

* see text

Table 4-14: Ethernet Voice Capacity at $\varphi = 1\%$. Bandwidth = 3, 10, 100 Mb/s Without silence suppression. $G_d = 0\%$. $V = 64$ kb/s

Increasing bandwidth, C, to increase the voice capacity of the network results in a decrease in T_p and consequently an increase in $a = \tau_p/T_p$. Since the maximum throughput of the Ethernet protocol is inversely related to the parameter a, absolute throughput increases less than linearly with increase in C. Thus, increasing C beyond a point may not be useful with the Ethernet protocol. This is seen clearly in Table 4-14 which gives the maximum number of voice stations that can be accommodated under various conditions if the maximum allowable loss is 1%. The voice coding rate is 64 kb/s, silence suppression is not used, and there is no data traffic. The values for 3 Mb/s are derived from the measurements reported in Section 4.5.1 by linear scaling from 105 kb/s to 64 kb/s. The values for 10 and 100 Mb/s are from simulation. For moderate values of D_{max}, i.e., 20 ms, the capacity is appreciable at $C = 3$ and 10 Mb/s. At 100 Mb/s, however, the capacity is a small fraction of the bandwidth. Only at large values of D_{max}, acceptable mainly when the call is restricted to the local area network, is the capacity at 100 Mb/s substantial. It is desirable to be able to handle calls to remote stations over the public telephone networks, performance at large D_{max} is less important and we consider only 10 Mb/s Ethernets in the subsequent discussions.

4.5.3.1. Voice Measures

Considering first the effect of the maximum allowable voice delay, D_{max}, in Figure 4-26, loss, φ, is plotted as a function of the number of voice stations. The data traffic offered load, G_d, is 20% and silence suppression is used. With $D_{max} = 2$ ms, loss occurs even with a single voice station. This is due to the short packet length, less than 16 bytes, and consequently the high probability of collision. With $D_{max} = 20$ and 200 ms, loss is close to zero for low values of N_v. As N_v increases beyond some value, loss begins to increases rapidly, causing a well-defined knee in the curve. Assuming a maximum acceptable loss level of 1%, the system voice capacity, $N_{v_{max}}^{(1)}$, is given in Table 4-15.

Comparing the curve in Figure 4-26 to the measured curves in Figure 4-24, we see that in the measured curves the knee is more well-defined and the curves are steeper in the region of loss. This is attributed to two factors, the effects of data traffic and its variations with time, and the effect of variations in the number of active voice calls due to the use of silence suppression. Curves for systems without silence suppression at 10 Mb/s are similar.

From Table 4-15, it is seen that the Ethernet is essentially unusable for voice traffic with $D_{max} = 2$ ms and $G_d = 20\%$ even if losses of 5% are tolerable. At $D_{max} = 20$ms, however, the capacity is substantially higher. Note that the maximum number of voice stations that could be accommodated on a 10 Mb/s channel in the absence of data traffic

$\varphi =$	1%	2%	5%
D_{max} = 2 ms			
without silence suppression	1	3	6
with silence suppression	2	6	14
D_{max} = 20 ms			
without silence suppression	43	51	62
with silence suppression	103	123	155
D_{max} = 200 ms			
without silence suppression	117	126	139
with silence suppression	259	300	351

Table 4-15: 10 Mb/s, 1 km Ethernet: Voice Capacity, D_{max} = 2, 20, 200 ms. With and without silence suppression. G_d = 20%.

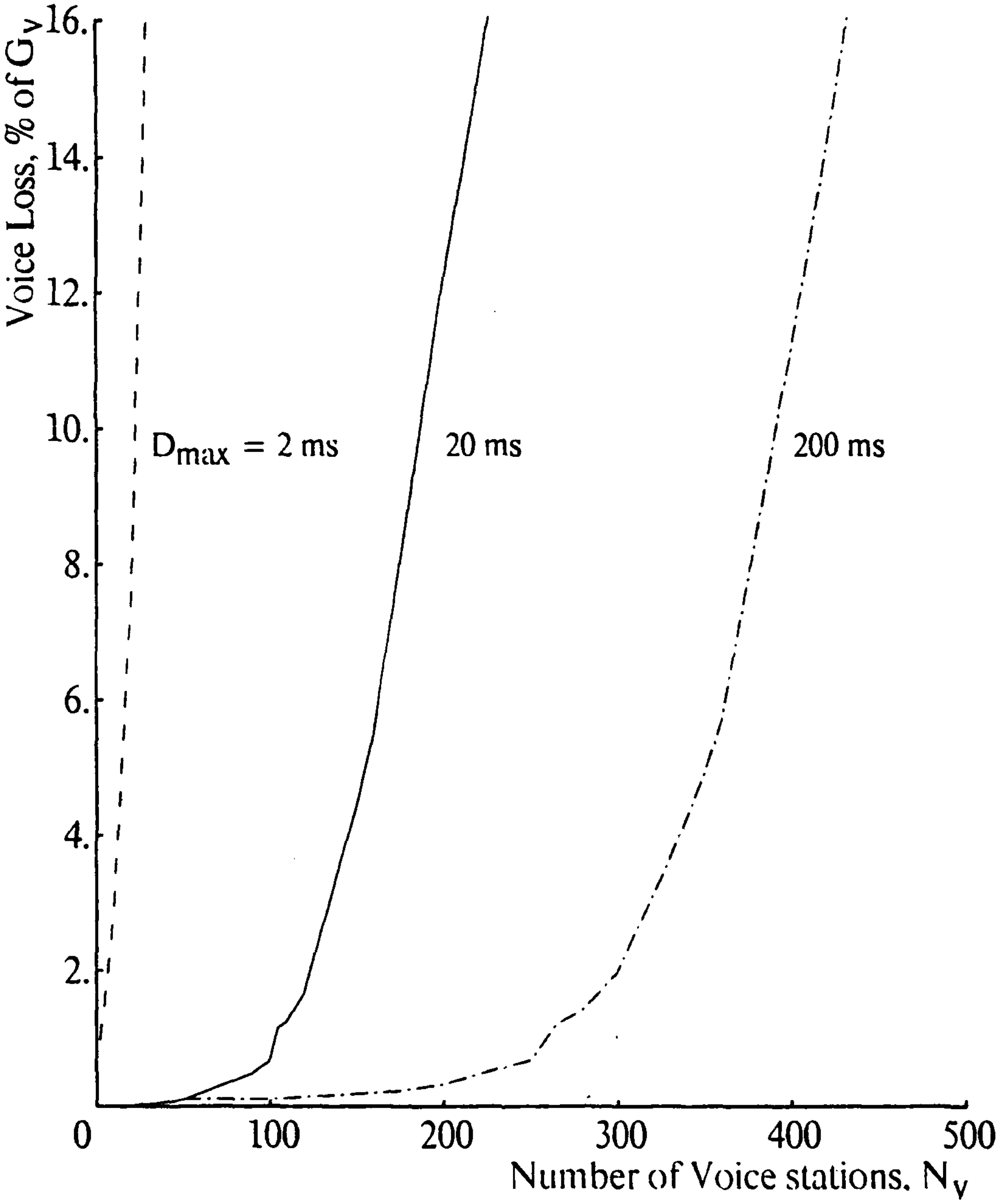


Figure 4-26: 10 Mb/s, 1 km Ethernet: Loss vs. N_v.
With silence suppression. G_d = 20%.

and overhead is $\lfloor C/V \rfloor$ = 156, without silence suppression. Allowing for the 20% data load, the Ethernet achieves only about 35% of the potential voice capacity with D_{max} = 20 ms. With D_{max} = 200 ms, the capacity is close to the maximum owing to the reduction in contention overhead with the longer voice packets. There is, thus, a strong incentive to design systems with a higher value of D_{max}. Silence suppression has little effect on voice capacity other than to increase it by a factor of 2.2 - 2.5 compared to the case without silence suppression. This is close to the ratio of the average talkspurt length to the average silence length, equal to 2.5.

To determine the effects of data traffic on voice performance, we keep D_{max} fixed at 20 ms and determine $N^{(\varphi)}_{v_{max}}$ for G_d = 0, 20 and 50% (Table 4-16). In the absence of data traffic, $N_{v_{max}}$ is fairly high, 120 compared to the maximum of 156. When data traffic is increased to 20%, equivalent to 30 voice stations, the voice capacity drops by nearly twice that number. When G_d is further increased to 50%, the voice capacity drops to zero. Thus, the lack of priority in the basic Ethernet protocol is seen to render voice traffic very susceptible to interference from data traffic (Table 4-17).

	φ =	1%	2%	5%
G_d = 0%				
	without silence suppression	119	119	122
	with silence suppression	257	266	281
G_d = 20%				
	without silence suppression	43	51	62
	with silence suppression	103	123	155
G_d = 50%				
	without silence suppression	0	0	12
	with silence suppression	0	1	30

Table 4-16: 10 Mb/s, 1 km Ethernet: Voice Capacity, G_d = 0, 20, 50%. With and without silence suppression. D_{max} = 20 ms.

Next, we examine the nature of the loss. In Table 4-18, details of the clip lengths and

G_d %	Without silence suppression			With silence suppression		
	η_v	η_d	η	η_v	η_d	η
0	76.4	0.0	76.4	65.4	0.0	65.4
20	27.5	20.0	47.5	25.9	20.0	45.9
50	0.0	50.0	50.0	0.0	50.0	50.0

Table 4-17: 10 Mb/s, 1 km Ethernet: Throughputs: voice, data and total. With and without silence suppression. D_{max} = 20 ms.

inter-clip times are presented when the total loss is 1% with G_d = 20% and a range of values for D_{max}. Corresponding statistics for φ = 5% are in Table 4-19. Recall that studies have indicated that for a given loss level, short clips occurring frequently are subjectively less of an annoyance than long clips occurring less frequently (Section 2.2.1.2) [Gruber & Strawczynski 85]. The threshold at which loss begins to be perceptible as lost syllables rather than background noise is about 50 ms [Campanella 76]. Under these criteria, it is seen to be desirable to set D_{max} as low as possible. For example, with D_{max} = 2 ms, the average clip length is less than 1 ms and the maximum about 6 ms, both well below the 50 ms threshold. With D_{max} = 20 ms, the average clip length is still fairly low, less than 10 ms. The standard deviation is high and the maximum greater than 100 ms. With D_{max} = 200 ms, both the mean and maximum clip lengths are well above the 50 ms threshold. Thus there is a trade-off in the selection of D_{max}. Smaller values lead to improved loss characteristics and allow a larger delay margin for transmission over other networks while. Larger values lead to a higher voice capacity due to reduction in overhead per bit and to lower collision rates.

Owing to the choice of P_{min} such that D_{min} is $0.8D_{max}$, for D_{max} = 20 and 200 ms, voice delay is constrained to lie in the range $0.8D_{max} < D_v \leq D_{max}$. As N_v increases, some voice packets get longer due to congestion delays. Others are successful with no or few collisions and are delayed only due to packetization. The average delay even under heavy traffic conditions with φ well above the acceptable limit of 1% is about 170 ms with D_{max} = 200 ms and G_d = 20% while the standard deviation is about 1/10th the average. Thus, in the region of interest, i.e., φ < 1%, $D_v \approx D_{max}$ and standard deviation is less than 5% of the average. This is a result of the parameters of the packetization protocol.

	$N_{v_{max}}^{(l)}$	Clip Length, T_l (ms)			Inter-clip time, T_{dt} (s)	
		Mean	Std dev	Max	Mean	Std dev
D_{max} = 2 ms						
without suppression	1	0.7	0.7	5.6	0.08	0.08
with suppression	2	0.8	0.8	5.0	0.08	0.08
D_{max} = 20 ms						
without suppression	43	6.9	10.9	134	0.72	0.70
with suppression	103	7.5	14.7	176	0.75	0.75
D_{max} = 200 ms						
without suppression	117	100.6	66.3	286	10.0	
with suppression	259	93.8	71.3	349	9.4	

Table 4-18: 10 Mb/s, 1 km Ethernet: Clipping Statistics at φ = 1%. With and without silence suppression. D_{max} = 2, 20, 200 ms. G_d = 20%

	$N_{v_{max}}^{(l)}$	Clip Length, T_l (ms)			Inter-clip time, T_{dt} (s)	
		Mean	Std dev	Max	Mean	Std dev
D_{max} = 2 ms						
without suppression	6	1.4	3.1	90	0.04	0.03
with suppression	14	1.5	4.2	136	0.03	0.03
D_{max} = 20 ms						
without suppression	62	15.4	27.4	238	0.34	0.32
with suppression	155	16.9	31.0	284	0.32	0.31
D_{max} = 200 ms						
without suppression	139	134.8	79.8	551	2.66	2.63
with suppression	351	126.3	78.3	498	2.10	2.21

Table 4-19: 10 Mb/s, 1 km Ethernet: Clipping Statistics at φ = 5%. With and without silence suppression. D_{max} = 2, 20, 200 ms. G_d = 20%

4.5.3.2. Data Measures

Having examined voice traffic performance and the impact of data traffic on voice performance, we now consider data traffic performance and the effects of voice traffic on it. In Figure 4-27, data throughput, η_d, is plotted as a function of the number of voice stations, N_v, for a 10 Mb/s, 1 km Ethernet. Silence suppression is not in effect. With data offered load, G_d, of 20%, curves are plotted for maximum voice delay, D_{max}, equal to 2, 20 and 200 ms. Also shown is a curve with $G_d = 50\%$ and $D_{max} = 20$ ms.

Comparing the curves for $G_d = 20\%$, it is seen that data throughput decreases with increase in N_v. This occurs because there are no packet-level priorities and the number of data stations is fixed. Thus, as N_v increases, the rate of voice-packet arrivals increases relative to the rate of data-packet arrivals. For a given N_v, with larger D_{max}, voice packets suffer fewer collisions and hence lower loss. Thus, the bandwidth available for data traffic is reduced compared to the situation with smaller D_{max}. Comparing the curves for $D_{max} = 20$ ms and $G_d = 20\%$ and 50%, the trends are similar. With silence suppression, similar effects are noted, except that the curves for different values of D_{max} and the same G_d are closer together.

Delay

The delay characteristics of interactive and bulk data traffic are similar and hence we discuss only the interactive traffic here. In Figure 4-28, average delay is plotted as a function of N_v for several values of D_{max}. $G_d = 20\%$ and silence suppression is not used. We note that the curves with silence suppression are very similar. On each curve, the point at which φ reaches 1% is marked with a circle. Initially, as N_v is increased, delay increases rapidly. Once loss begins to occur, the increase in voice traffic with further increase in N_v is reduced and the increase in delay is much more gradual. Note that delay is not noticeably dependant on D_{max}. Throughout the range shown, standard deviation of data delay is about 2 to 3 times the average, ranging occasionally up to 5 times the average.

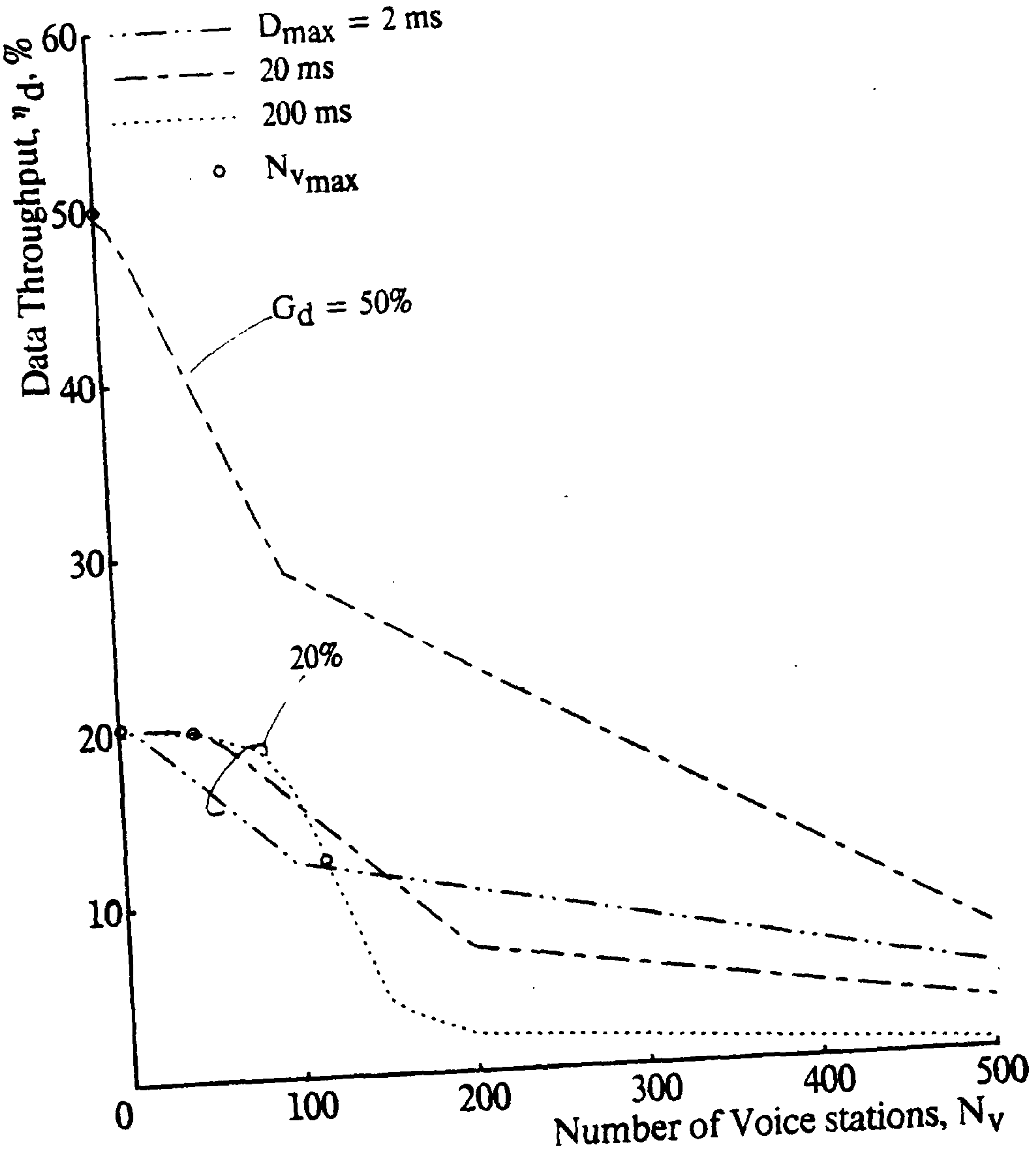


Figure 4-27: 10 Mb/s, 1 km Ethernet: Data Throughput vs. N_v. Without silence suppression. Parameters: D_{max}, G_d

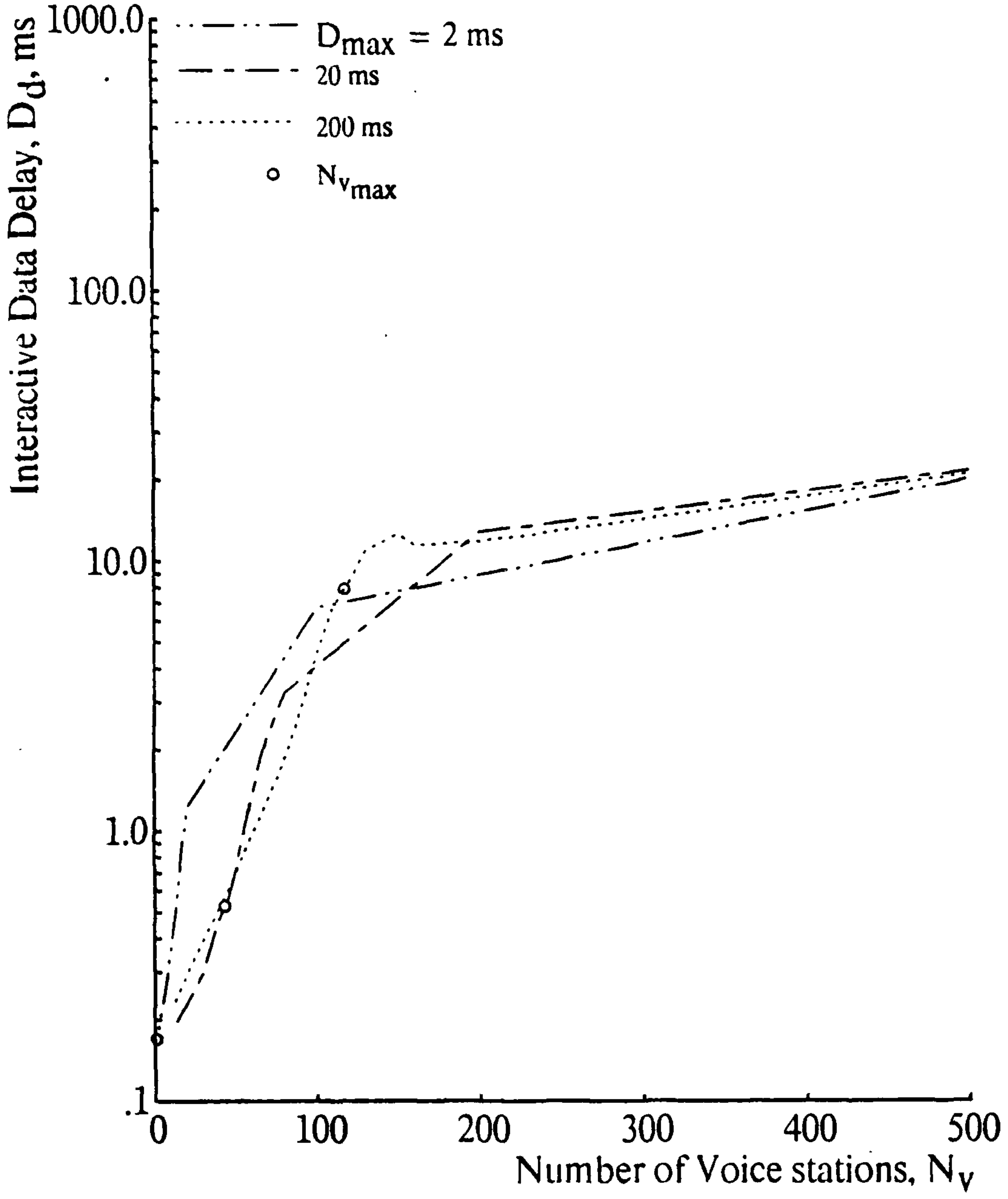


Figure 4-28: 10 Mb/s, 1 km Ethernet: Interactive Data Delay vs. N_v. Without silence suppression. G_d = 20%. Parameter: D_{max}.

4.5.4. Discussion

We have used measurements on a 3 Mb/s Ethernet with emulated voice traffic to show that the contention-based random-access protocol can provide adequate service to stream-based real-time traffic. The network capacity is 30-40 simultaneous voice conversations with $V = 64$ kb/s. This corresponds to a utilization of about 90% of the bandwidth. We have then extended this work to integrated voice/data traffic at a higher bandwidths via simulation. Owing to the higher propagation delay relative to the packet transmission time, the use of the Ethernet for real-time traffic is more restricted at 10 Mb/s. With $D_{max} > 20$ ms, moderate to high utilization is obtained. At lower values, however, utilization drops considerably. When bandwidth is increased to 100 Mb/s, D_{max} must be on the order of 200 ms to obtain adequate utilization. The interactions of voice and data traffic have been quantified over a wide range of parameters. An empirical optimization of the parameter P_{min} of our voice packetization protocol indicates that the optimum value of P_{min} as a fraction of P_{max} decreases as P_{max} decreases.

4.6. Summary

We have accurately characterized the performance of the Ethernet protocol via measurements on operational networks and simulation. Measurements on 3 and 10 Mb/s Ethernets with artificially-generated data traffic loads indicate that the protocol performs well when the packet transmission time is large compared to the propagation delay, with throughput greater than 97% of capacity. On a 10 Mb/s network, with 64 byte packets, however, performance is poor, with η being about 25%. By measuring delay distributions we have shown that while individual packet delays can be large under heavy loads, the variation is high and most packets suffer relatively modest delays.

A comparison of our measurements with the predictions of analytical models of CSMA/CD from the literature indicated discrepancies, especially for large a. This led to a study, via simulation, of the performance of the Ethernet at large a. We have shown that a modification to the retransmission algorithm enables higher throughput than that of the standard algorithm to be achieved, especially with large numbers of stations. Since the throughput of the modified algorithm is close to that predicted by analytic studies with optimum assumptions, we surmise that the modified algorithm is near-optimal. This study enabled us to determine the regions of validity of the use of some analytical models of CSMA/CD from the literature for the prediction of Ethernet performance. Using simulation we have also studied the effects of different distributions of stations on linear bus Ethernets. Stations at the ends and stations in small clusters were shown to achieve poorer performance relative to the others.

Using measurements on a 3 Mb/s Ethernet, we have shown that voice traffic can be supported under acceptable constraints despite the random nature of the access protocol. This is due in part to our new variable-length packet voice protocol. Simulation of voice/data traffic at higher bandwidths and under a wide range of parameters indicates the trade-offs between voice capacity on the one hand and, on the other, maximum voice delay and the quality of the voice signal. Data traffic is shown to have an adverse impact on voice capacity. In the desirable region of operation, i.e., when voice loss is low, voice traffic has minimal effect on data throughput. The effect of variation of the voice-packetization protocol parameter P_{min} has been investigated over a range of conditions. The optimum is found to decrease from about $0.8P_{max}$ with D_{max} = 200 ms to $0.4P_{max}$ with D_{max} = 2 ms.

Chapter 5
Token Bus

The Token Bus protocol utilizes explicit token-passing to achieve round-robin scheduling on a single broadcast bus [IEEE 85b]. This helps overcomes the inefficiency and high variance of delay caused by collisions in the CSMA/CD protocol. The Token Bus protocol is fair and guarantees an upper bound on delay, provided that packet lengths are bounded. Priorities are incorporated by means of timers to limit the maximum token rotation time and the time that a station may hold the token. The disadvantage is an increase in complexity of the protocol especially to handle error conditions such as loss of the token. In this Chapter, we study the performance of a Token Bus protocol for integrated voice data traffic. The protocol is similar to the IEEE 802.4 Token Bus Standard. The protocol is described in the next section with differences from the 802.4 standard being identified. Next, the effect of the priority parameter, the token rotation timer used by data stations, is examined. This is followed by an investigation of several variants of the protocol with only voice traffic. Finally, the performance of the protocol with voice/data traffic within the framework of Chapter 2 is systematically characterized.

5.1. Token Bus Protocol

The physical topology of the Token Bus is similar to that of the Ethernet, i.e., a broadcast bus with stations connected by means of passive taps. The protocol, however, is quite dissimilar. During normal operation, a single logical token exists on the network. A station may transmit a packet only when it has possession of the token. After transmission, the token is passed on to the succeeding station in a *logical ring* (Figure 5-1). Thus, contention-free operation is achieved. Note that at a given point in time, the logical ring may contain only a subset of all the stations on the network. The complete protocol

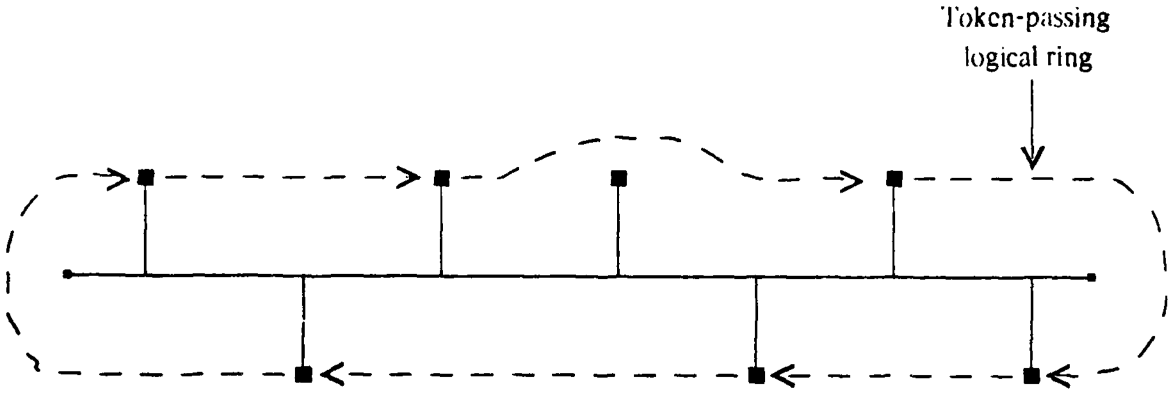


Figure 5-1: Token Bus Topology

includes procedures for the handling of error conditions such as the loss of the token, duplicate tokens, and entry of stations to and exit from the logical ring. We assume that such conditions occur infrequently and hence do not discuss them. The IEEE 802.4 standard is a complete specification [IEEE 85b].

One of the determinants of performance is the ordering of stations in the logical ring. If the ordering corresponds to the physical order of the stations on the bus propagation delay incurred in passing the token is minimized. This is particularly beneficial at high bandwidths when the packet transmission time is small, i.e., when a is large. In practice, however, over time the logical ordering is likely to change as stations are moved between locations, are added or are removed, thus increasing the propagation delay incurred in passing the token. In the absence of a mechanism for ensuring optimum ordering, we assume that the logical ring is constructed by choosing stations at random. This results in an average propagation delay of $\tau_p/3$ between any pair of stations. In Section 5.2 we investigate some ramifications of this assumption.

The IEEE 802.4 standard allows a station to transmit several packets during possession of the token, limited either by a maximum token holding timer for *synchronous* traffic, or by a limit on the time since the previous reception of the token by the station for *asynchronous* traffic. The token is then passed in a separate packet. In our voice protocol, a voice station transmits all the accumulated samples when it gains access to the network. Thus, it transmits only one packet during each round. Likewise, by our assumption of a single buffer per data station, data stations too transmit only one packet per round. As an optimization, we assume that the token is piggy-backed on the single packet transmitted by a station. The improvement achieved by this is also studied in Section 5.2. Note that a station that does not have a packet to transmit must still transmit a packet, consisting only of a header, to pass the token on to its successor.

5.1.1. The Priority Mechanism

We distinguish two priority classes, data and voice. Voice is considered the higher priority. Each voice station transmits at most one packet per round with the packet data length limited to $P_{max} = D_{max}/V$. Thus, the *token holding time* of a voice station is limited to $(P_{max} + P_{ov})/C$, where P_{ov} is the total overhead per packet. Data stations are restricted by the *token rotation timer (TRT)* mechanism. A data station may transmit only if the time since the previous reception of the token is less than the priority parameter, TRT. In this Section, we examine by simulation the effects of various values of TRT. We note that the IEEE 802.4 standard distinguishes 3 classes of asynchronous traffic with a separate TRT specified for each class and one class of synchronous traffic with a specified maximum token holding time.

We consider a 100 Mb/s, 5 km network and voice/data traffic as defined in Section 2.4 with $D_{max} = 20$ ms and $G_d = 20\%$. For TRT = 15, 20, 25 and ∞ ms, we plot voice loss as a function of N_v in Figure 5-2. The difference between the four cases is small. At small N_v, loss is zero in all cases. At large N_v, the curves for TRT $\leq$ 25 ms merge. In the region of interest, φ on the order of 1%, TRT of 25 ms yields the same voice performance as TRT of infinity.

In Figure 5-3, data throughput, η_d is plotted as a function of N_v for the four values of TRT. With finite TRT, η_d drops to zero when N_v crosses voice capacity. The value of N_v at which η_d reaches zero increases with TRT. With TRT = ∞, η_d decreases monotonically with increase in N_v because of the restriction of a single packet per station per round but does not drop to zero.

For any TRT $\leq D_{max}$, voice capacity is the same. Data throughput, however, drops to zero at lower values of N_v for lower values of TRT. Hence, to maximize voice capacity while minimizing the adverse impact on data throughput, we use TRT = D_{max} in the voice/data performance evaluation in Section 5.4. Note that in Chapter 7, we also use larger values of TRT to accord equal treatment to data traffic in the Token Bus and Expressnet.

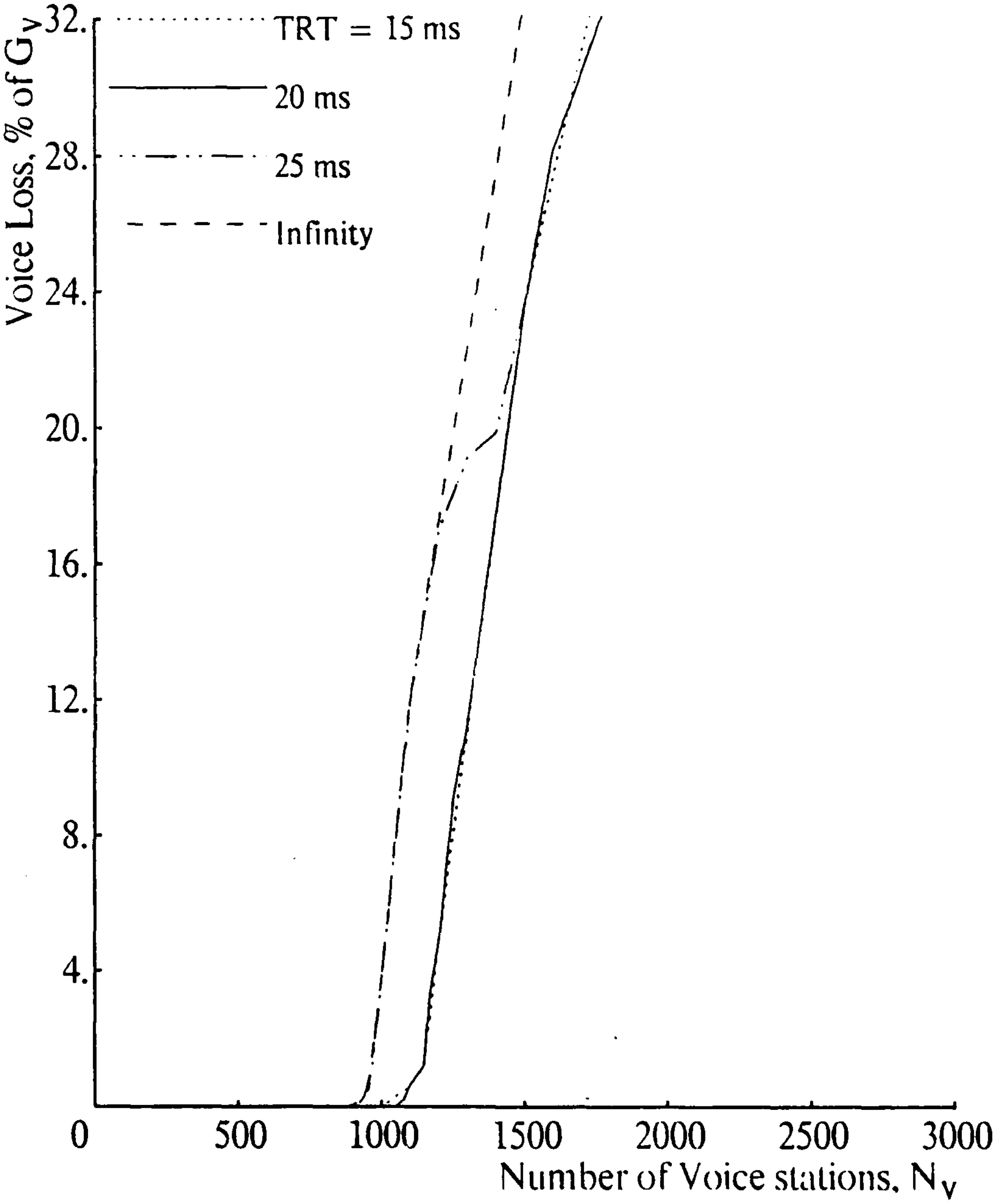


Figure 5-2: 100 Mb/s, 5 km Token Bus: Loss vs. N_v.
Token Rotation Timer, TRT = 15, 20, 25, ∞ ms.
D_{max} = 20 ms. G_d = 20%.

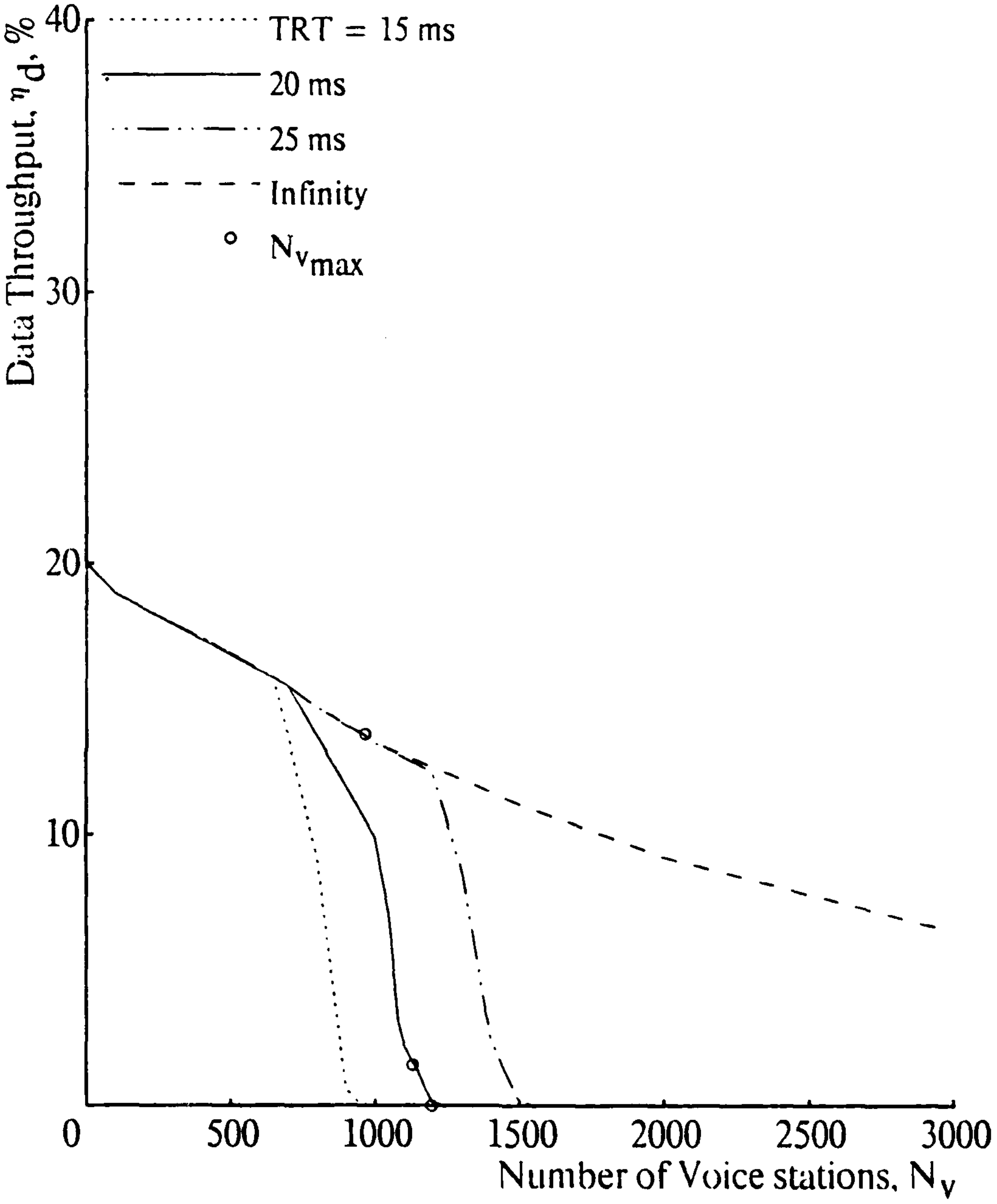


Figure 5-3: 100 Mb/s, 5 km Token Bus: Data Throughput vs. N_v.
Token Rotation Timer, TRT = 15, 20, 25, ∞ ms.
D_{max} = 20 ms, G_d = 20%.

5.2. Voice Traffic Performance

In the absence of data traffic and when silence suppression is not used, a Token Bus carrying only voice traffic achieves maximum performance when each station transmits maximum length packets at regular intervals. The operation of the system is deterministic and a simple analytic expression for the voice capacity, $N^{(0)}_{v_{max}}$, can be obtained following the method used in [Fine 85]. When $N_v = N^{(0)}_{v_{max}}$, the round length is D_{max} and the following equality holds:

$$D_{max} = (D_{max} V/C + t_p + t_o + T_{tok} + \tau) N_v$$

where t_p and t_o are the transmission times of the packet preamble and overhead respectively. T_{tok} is the total transmission time of the token packet, and τ is the mean propagation delay incurred in passing the token from one station to the next. Thus, after multiplication by C for conversion of transmission times to bits, the voice capacity with $\varphi = 0\%$ is given by:

$$N_v = D_{max} C/(D_{max} V + P_p + P_o + P_{tok} + \tau C) \qquad (5.1)$$

where P_p is the preamble length, P_o, the overhead length and P_{tok}, the token length including any overhead. By substitution of appropriate values for variables in the above equation, the capacity of several variants of the protocol may be obtained. First, for the IEEE 802.4 Token Bus, a separate packet is used for the token. The length of this packet is $P_{tok} = P_p + P_o + P_k$, where P_k is the length of the token. We assume that $P_k = 0$, i.e., an empty packet constitutes a token. In our study, we assume that each station transmits a single packet per round and that the token is implicitly passed in this packet. Thus, $P_{tok} = 0$.

In the optimum ordering of stations, the logical ring corresponds to the physical order of the stations on the bus. Thus, the total propagation delay incurred per round is twice the end-to-end propagation delay or $2\tau_p$. Hence, $\tau = 2\tau_p/N_v$. Under the assumption of random ordering of the stations in the logical ring, $\tau = \tau_p/3$.

We present in Table 5-1 the voice capacity calculated from Equation (5.1) for both the standard Token Bus and the protocol with piggy-backed tokens. In both cases, capacity is presented for the optimum and random orderings. The following parameter values are

	Optimum Order		Random Order	
D_{max}	Standard	Fast	Standard	Fast
10 Mb/s, 1 km ($\lfloor C/V \rfloor$ = 156)				
2 ms	32	53	31	51
20 ms	113	131	112	129
200 ms	150	153	150	153
100 Mb/s, 5 km ($\lfloor C/V \rfloor$ = 1562)				
2 ms	316	524	137	165
20 ms	1128	1309	768	848
200 ms	1504	1532	1416	1441

Table 5-1: Token Bus: Voice Capacity at $\varphi = 0\%$, $C = 10$ and 100 Mb/s. Separate and piggy-backed tokens. Optimum and random ordering. Without silence suppression, $G_d = 0\%$.

used: Network parameters: 10 Mb/s, 1 km and 100 Mb/s, 5 km. Note that for the 1 km network, $\tau_p = 5\ \mu s$ and for the 5 km network, $\tau_p = 25\ \mu s$. Voice station parameters: V = 64 Kb/s, D_{max} = 2, 20, 200 ms; P_p = 64 bits; P_o = 80 bits overhead + 100 bits gap. The packet data lengths corresponding to D_{max} = 2, 20 and 200 ms are 16, 160 and 1600 bytes respectively. The maximum capacity assuming ideal conditions is given by $\lfloor C/V \rfloor$ = 156 and 1562 stations for C = 10 and 100 Mb/s respectively.

Considering the piggy-backed versus separate token variants (labelled *Fast* and *Standard* respectively in the Table), at 10 Mb/s, the piggy-backed scheme is marginally better than the standard scheme. The difference is greater at small values of D_{max}. With D_{max} = 200 ms, the overhead of a separate token packet, 30 bytes, is small compared to the packet data length. With D_{max} = 2 and 20 ms is it significant. Thus, the simulation results that we present for the piggy-backed scheme later in this Chapter and in Chapter 7 can be applied to the IEEE 802.4 Bus with only a small degree of over-estimation.

Comparing the two orderings, optimum and random, we find that at 10 Mb/s the difference is negligible. Here, τ_p is much smaller than the packet transmission time, T_p. At 100 Mb/s, however, T_p decreases by an order of magnitude while τ increases by a factor of 5 due to the increase in network length. τ_p is now comparable to T_p, especially for smaller D_{max}. In the optimum ordering, the per packet propagation delay, $\tau = \tau_p/N_v$ while in the random ordering, $\tau = \tau_p/3$. Since $N_v \gg 3$, the optimum ordering is superior to the random ordering, especially at small D_{max}.

5.3. Minimum Voice Packet Length

Before we can proceed further, it is necessary to select near-optimal values of the minimum voice packet length, P_{min}. Owing to the orderly round-robin scheduling, the round length increases in proportion to the number of voice stations. There is little variation in round length and hence all voice packets are of similar length. In contrast to the Ethernet (Section 4.5.2), there is no reason to choose P_{min} large.[15] In this respect, the Token Bus is very similar to the Expressnet, also a round-robin scheduling protocol and the discussion of P_{min} with respect to the Expressnet holds here (Section 6.3). Hence, the values of P_{min} we use in the Token Bus evaluations are the same as in the Expressnet evaluations, i.e., 8 bytes for D_{max} = 20 and 200 ms, and 1 byte for D_{max} = 2 ms. These yield D_{min} = 1 ms and 0.125 ms.

5.4. Voice/Data Traffic Performance

We are now prepared to study the performance of the Token Bus with integrated voice/data traffic. Following the strategy used with the Ethernet in Section 4.5.3, we present first voice traffic measures and the effects of data traffic on voice and then data traffic measures. The traffic parameters are as summarized in Section 2.4 and Table 5-2. For reasons discussed above, we consider the piggy-backed token variant of the protocol and assume that stations in the logical ring are in random order.

[15] Based on simulation of a 10 Mb/s Token Bus with voice packet length varying between 6 and 12 ms, DeTreville arrived at a similar conclusion [DeTreville 84].

Network Parameters:	
Data token rotation timer, TRT	D_{max}
Voice token holding time, THT	$D_{max}V/C + P_o/C$
Station Parameters:	
Packet overhead, P_o	10 bytes
Packet preamble, P_p	64 bits
Carrier detection time, t_{cd}	1.0 μs at C = 10 Mb/s 0.1 μs at C = 100 Mb/s
Voice Traffic Parameters:	
Minimum delay, D_{min}	0.125 ms at D_{max} = 2 ms 1.0 ms at D_{max} = 20, 200 ms

Table 5-2: Simulation Parameters

5.4.1. Voice Measures

First, we consider the impact of the maximum allowable voice delay, D_{max}, on voice traffic performance. In Figure 5-4, voice loss is plotted as a function of N_v for various values of D_{max} for a 100 Mb/s, 5 km network with data offered load, G_d = 20%. Performance is shown with and without silence suppression. Considering the curves for silence suppression, it is seen that there is no loss until N_v reaches some value dependant on D_{max}. There is a knee above which loss increases rapidly, similar to, though sharper than, that observed in the Ethernet (Figure 4-26). The voice capacity at D_{max} = 2 ms is negligible. There is a substantial increase when D_{max} is increased to 20 ms. The poor performance at D_{max} = 2 and 20 ms is attributed to the propagation delay incurred in passing the token. This is $\tau_p/3$ on the average due to the assumption of random ordering of stations in the logical ring. At D_{max} = 2 ms, the packet overhead of 10 bytes is significant compared to the 16 bytes of voice samples per packet. When D_{max} is further increased to 200 ms, there is a further large increase in voice capacity. The throughput is then about 80% of the bandwidth.

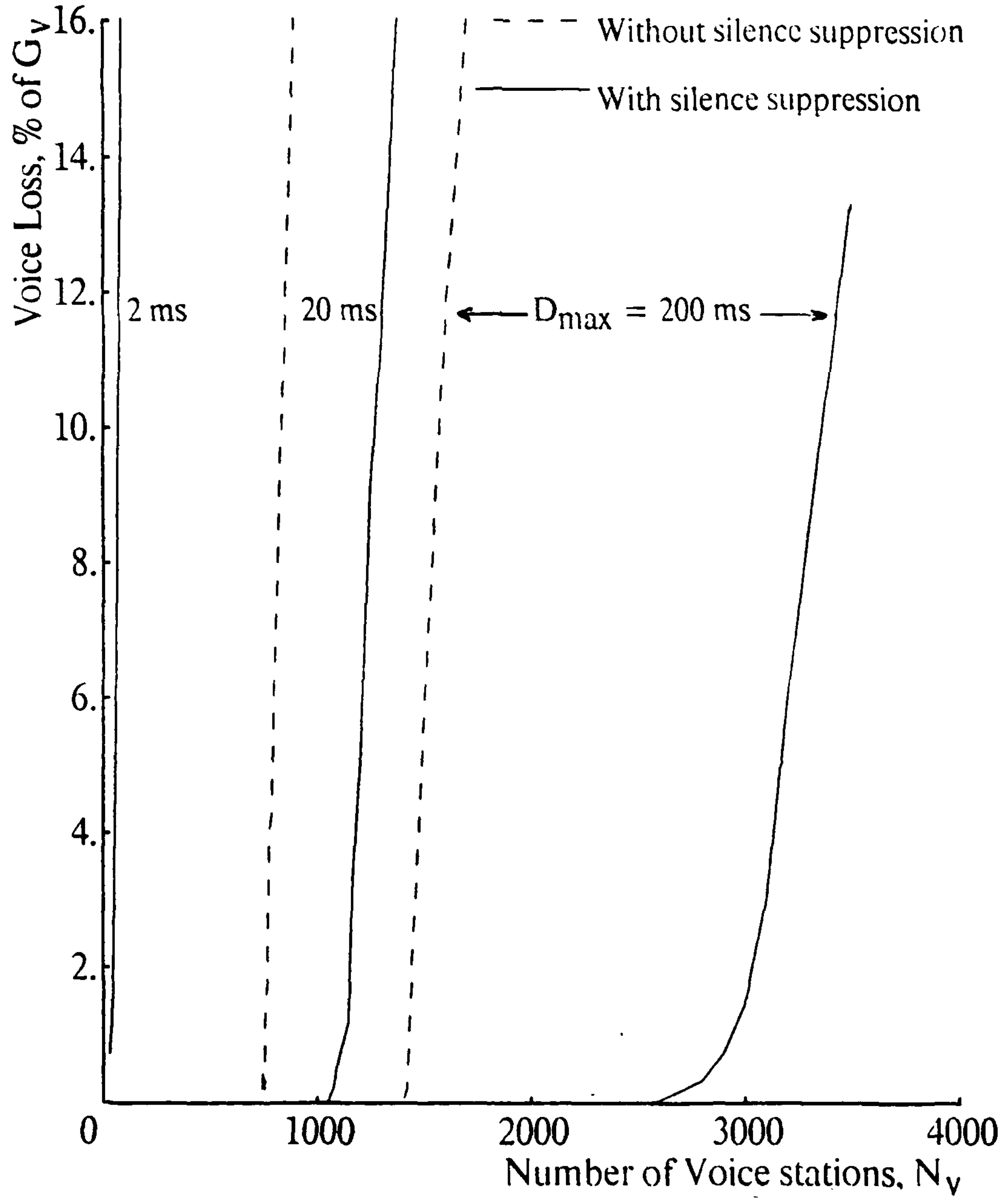


Figure 5-4: 100 Mb/s, 5 km Token Bus: Loss vs. N_v. G_d = 20%. Parameter, D_{max}.

	10 Mb/s, 1 km			100 Mb/s, 5 km		
φ =	1%	2%	5%	1%	2%	5%
D_{max} = 2 ms						
With suppression	11	21	27	15	47	56
D_{max} = 20 ms						
Without suppression	103	108	122	757	770	790
With suppression	207	224	241	1132	1157	1200
D_{max} = 200 ms						
Without suppression	152	154	159	1430	1443	1490
With suppression	339	346	364	2936	3035	3167

Table 5-3: Token Bus: $N_v^{(\varphi)}$ for D_{max} = 2, 20, 200 ms.
C = 10, 100 Mb/s. G_d = 20%.

The curves for performance without silence suppression are similar though the knee is at a correspondingly lower value of N_v. In Table 5-3, $N_{v_{max}}^{(\varphi)}$ is shown for φ = 1, 2 and 5% for values of parameters corresponding to Figure 5-4. At 10 Mb/s, the increase in capacity achieved by the use of silence suppression is about a factor of 2 compared to the decrease by a factor of 2.5 in the bandwidth required per station. When silence suppression is used, stations are assumed to remain in the logical ring even when in the silent state. Thus, they contribute delay in passing the token. The effect is more pronounced at 100 Mb/s when the propagation delay is larger compared to the packet transmission time, the relative increase in capacity achieved by the use of silence suppression being only about 1.5.

The effect of propagation delay can be seen by comparing capacity at the two bandwidths with other parameters constant. The increase in capacity is less than the relative increase in bandwidth. The relative increase approaches the ratio of bandwidths as D_{max} is increased. We note that the poor performance at D_{max} = 2 and 20 ms and C = 10 Mb/s is due almost entirely to packet overhead. For the parameters used, propagation delay is a significant factor only at the higher bandwidth.

	10 Mb/s, 1 km			100 Mb/s, 5 km		
φ =	1%	2%	5%	1%	2%	5%
G_d = 0%	232	237	248	1197	1212	1260
G_d = 20%	207	224	241	1132	1157	1200
G_d = 50%	197	222	237	1008	1034	1082

Table 5-4: Token Bus: Voice Capacity, G_d = 0, 20, 50%.
With silence suppression. C = 10, 100 Mb/s. D_{max} = 20 ms.

The impact of various data loadings on voice traffic performance is minimal. Owing to the choice of TRT = D_{max}, when N_v reaches a value such that loss occurs, the round length exceeds D_{max} and data throughput drops to zero regardless of G_d and other parameters. This is summarized in Table 5-4.

Turning next to the nature of the loss suffered by voice stations, we find that due to the orderly round-robin scheduling, mean clip lengths are low, standard deviation is low and the clips occur very regularly (Table 5-5). This is the desired mode of loss (Section 2.2.1.2). The low standard deviation of the inter-clip time is due to the almost constant round lengths. Thus, every packet suffers an equal clip in every round. Note that the standard deviations increase somewhat when silence suppression is used. For a given loss level, e.g., φ = 1%, the mean clip length increases with D_{max}. This occurs because loss occurs when the round length is approximately equal to D_{max} at the rate of 1 clip per round. With larger D_{max}, the rate of clipping decreases and the mean clip length must increase proportionately for a constant loss level.

5.4.2. Data Measures

We now turn to a consideration of data traffic performance and the effect of voice traffic on it. As was indicated above, owing to the TRT priority mechanism, when the number of voice stations reaches a value such that the round length exceeds TRT = D_{max}, data throughput drops to zero. This is seen in Figure 5-5 which shows the variation of η_d with N_v for various values of D_{max} and G_d for a 100 Mb/s network with silence

	$N_{v_{max}}$	Clip Length, T_l (ms)			Inter-clip time, T_{ilt} (s)	
		Mean	Std dev	Max	Mean	Std dev
D_{max} = 2 ms						
with suppression	15	0.05	0.04	0.1	0.004	0.002
D_{max} = 20 ms						
without suppression	757	0.20	0.06	0.2	0.020	0.000
with suppression	1132	0.30	0.31	2.8	0.031	0.023
D_{max} = 200 ms						
without suppression	1430	2.10	0.09	2.9	0.210	0.073
with suppression	2936	10.60	11.74	44.4	1.060	

Table 5-5: 100 Mb/s, 5 km Token Bus: Clipping Statistics at $\varphi = 1\%$. $G_d = 20\%$. $D_{max} = 2, 20, 200$ ms.

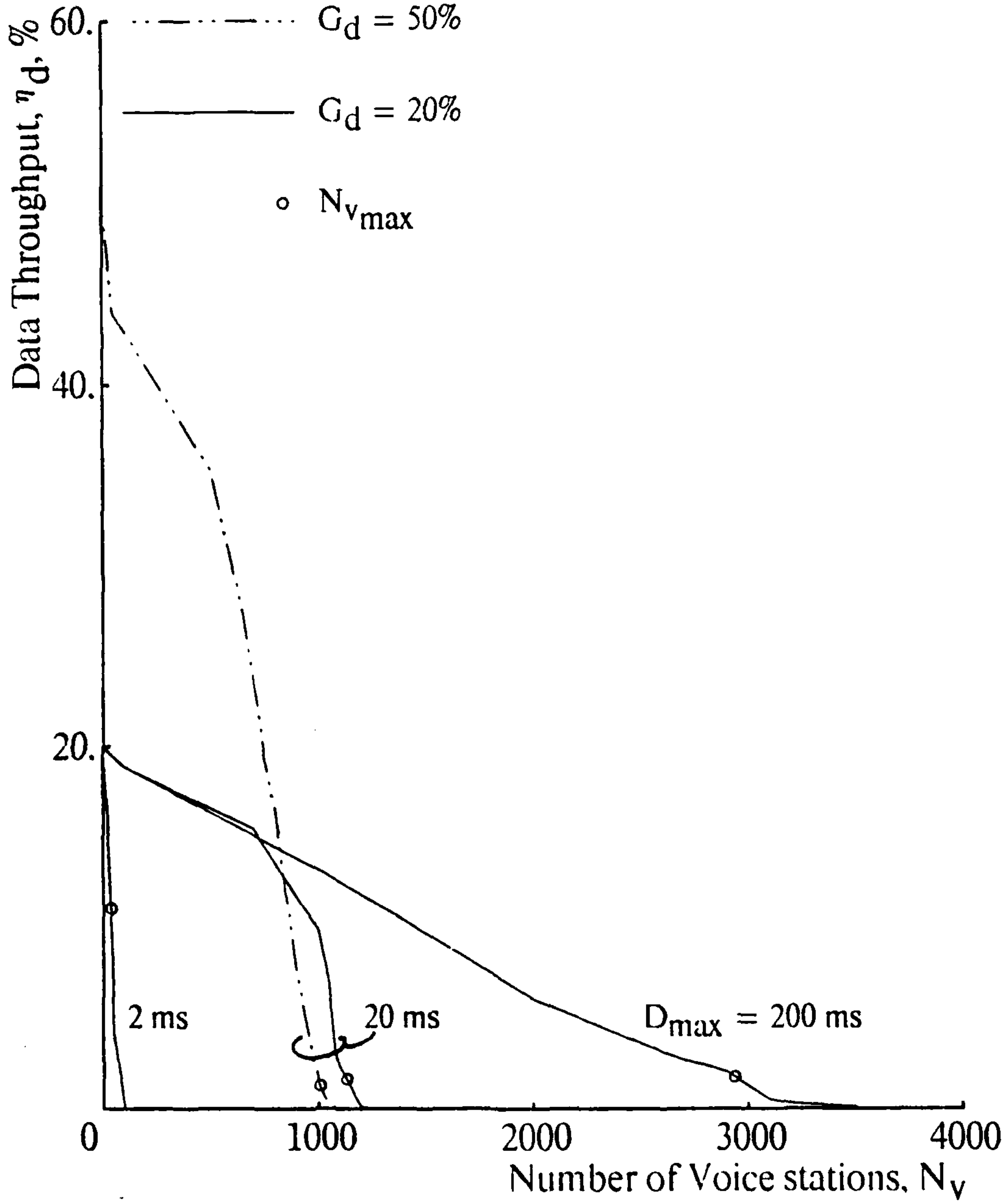


Figure 5-5: 100 Mb/s, 5 km Token Bus: Data Throughput vs. N_v. With silence suppression. G_d = 20, 50%. Parameter, D_{max}.

with N_v for various values of D_{max} and G_d for a 100 Mb/s network with silence suppression. Similar behaviour is observed with other parameter values. η_d decreases with increase in N_v. As N_v approaches $N^{(1)}_{v_{max}}$, the rate of decrease increases. Note that by using TRT $\geq D_{max}$, we can ensure higher data throughput at voice capacity (see Section 5.1.1).

As is to be expected from the throughput performance, data delay increases with N_v, growing without bound as N_v approaches the value at which η_d drops to zero (Figure 5-6). Standard deviation is approximately equal to the average over the whole range when silence suppression is in effect. When silence suppression is not used, owing to the more deterministic voice traffic, standard deviation is much lower than the average.

5.5. Summary

In this Chapter we have presented a study of several aspects of performance of a token-passing bus local area network. The protocol considered is similar to the IEEE 802.4 standard. First we considered the performance of several variants with only voice traffic, obtained by a simple analytic expression for the case without silence suppression. The use of piggy-backed tokens was shown to cause a marginal increase in capacity, except at D_{max} = 2 ms where the increase is larger. Optimum ordering of the stations in the logical ring has little advantage relative to random ordering at 10 Mb/s. At 100 Mb/s, there is a substantial improvement. The effects of several values of the priority parameter, TRT, used for data traffic were presented.

With integrated voice/data traffic, the Token Bus protocol achieves good performance at 10 Mb/s. At 100 Mb/s, propagation delay begins to play a significant part and performance is poor under tight delay constraints. The token rotation timer priority mechanism is seen to be favourable for the higher priority traffic, voice. The throughput of the lower priority traffic, data, drops to zero as the number of voice stations increases above $N^{(0)}_{v_{max}}$. We note that other priority schemes can be implemented on a Token Bus. In particular, the alternating round scheme presented for the Expressnet in the next Chapter is compared with the TRT mechanism in Chapter 7.

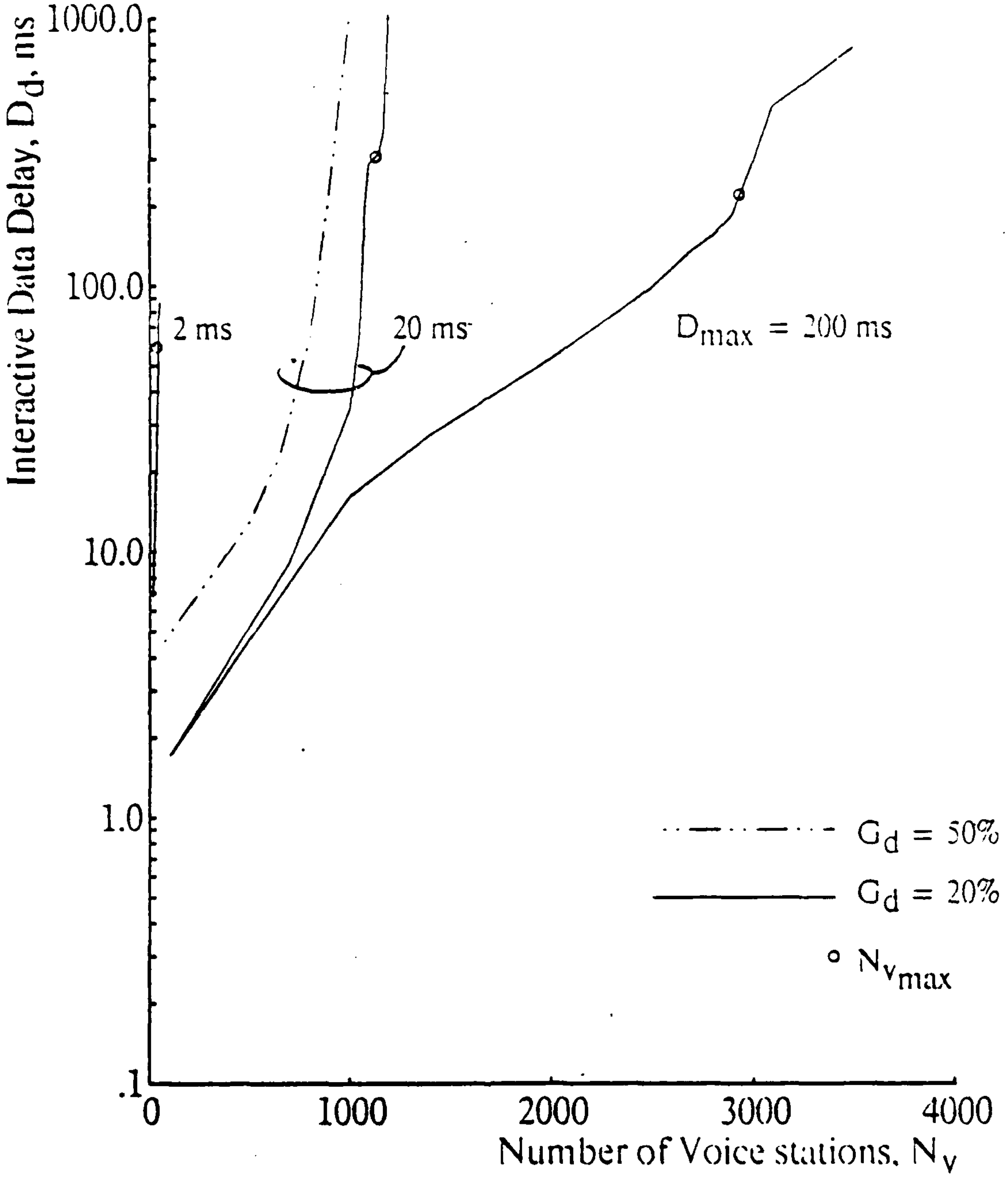


Figure 5-6: 100 Mb/s, 5 km Token Bus: Interactive Data Delay vs. N_v. With silence suppression. G_d = 20, 50%. Parameter, D_{max}.

Chapter 6
Expressnet

The network protocols considered so far are the random-access CSMA/CD protocol and the round-robin Token Bus scheme. The former is attractive owing to its simplicity and is efficient at low to medium bandwidths. By the use of an explicit token, the latter provides good performance at higher bandwidths than CSMA/CD. The performance of the Token Bus has been shown to be strongly dependant on the order of token passing (Section 5.2). The optimum ordering enables high utilization to be achieved at high bandwidths but is difficult to maintain in a practical network. Another round-robin scheme, Expressnet, uses an implicit token-passing algorithm with inherently optimum ordering to achieve high utilization [Fratta *et. al.* 81, Tobagi *et. al.* 83]. In this Chapter, we describe the Expressnet protocol, study the effects of some protocol parameters, and study the performance of the Expressnet within our voice/data framework. We note that the Expressnet is very similar to several of the DAMA schemes and that our results are therefore indicative also of the performance of these other schemes [Fine & Tobagi 84].

6.1. Expressnet Protocol

The Expressnet uses a *folded uni-directional bus* structure to achieve broadcast operation with conflict-free round-robin scheduling (Figure 6-1) [Fratta *et. al.* 81, Tobagi *et. al.* 83]. Each station has three taps, a receive tap on the in-bound bus and transmit and carrier sense taps on the out-bound bus. Note that the sense tap is upstream of the transmit tap. A packet transmitted on the out-bound bus by any station propagates over the connecting link and down the entire in-bound bus. Any station can receive the packet from the in-bound bus, achieving broadcast operation.

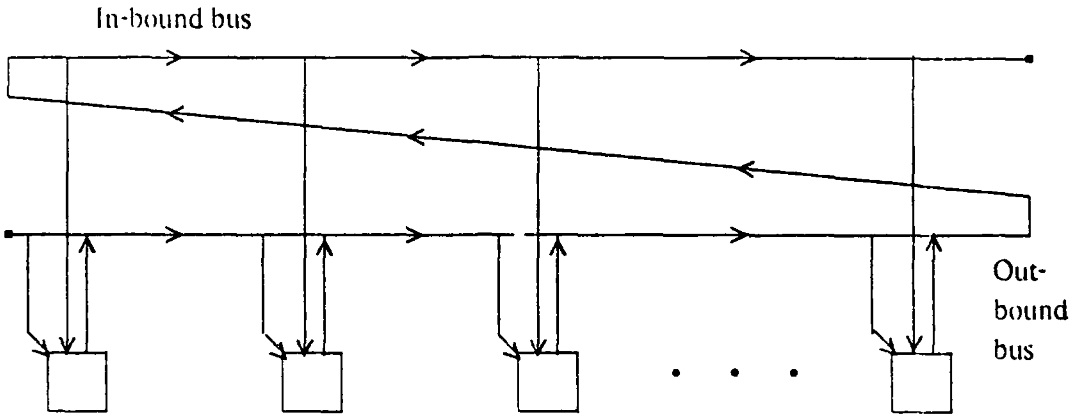


Figure 6-1: Expressnet: Folded-bus topology

In order to achieve fully distributed round-robin scheduling, the end of each packet transmission on the out-bound bus, *EOC(out)*, is used as a synchronizing event. Assume that station i has just transmitted a packet. The event EOC(out) emanates from i and propagated down the out-bound bus. Any backlogged station, j, downstream of i senses this event and starts to transmit immediately. While transmitting, j monitors the out-bound bus for transmissions from any upstream stations. If such a transmission is detected, j aborts its attempt. Thus, of all the statiรns that attempt to transmit, the most upstream one is successful. There may be a period of overlap at the start of the packet, on the order of t_{cd}, the time to detect carrier. The end of this new packet forms the next synchronizing event. Note that once a station has transmitted a packet it does not receive the EOC(out) event again and hence can transmit at most one packet per round. Transmissions within each round are ordered by station location without the stations needing to have any knowledge of their respective locations. We note that this protocol is an example of the *attempt-and-defer* sub-class of the DAMA protocols [Fine & Tobagi 84].

The succession of packets in a round form a *train*. At the end of a train, a mechanism is necessary to start the next train. Note that the gap between successive packets in a train is t_{cd}, the carrier sense time. Thus, any station can detect the end of a train when an idle period of $2t_{cd}$ elapses after the end of a packet on the in-bound bus. This event, *EOT(in)*, visits each station in order and forms the synchronizing event for start of the new round. All backlogged stations detecting EOT(in) immediately start transmitting, following the attempt-and-defer strategy described above. Between successive trains there is a gap equal to $2\tau_p + t_{cd}$ for the end of train to propagate from the out-bound to the in-bound bus and be detected. Under heavy traffic with N stations, the propagation delay overhead per packet is $2\tau_p/N$. Thus, the Expressnet operates efficiently with large N even when τ_p is large relative to the packet transmission time, T_p.

The Expressnet protocol includes mechanisms for keeping the net *alive* even when all stations are idle, and for *cold-start* when the network is powered up. These are not germane to our study and hence we do not describe them.

6.1.1. A Priority Mechanism

A simple priority mechanism is to allocate rounds for particular traffic types [Tobagi *et. al.* 83]. For example, in a voice/data context, alternate rounds can be allocated to each of the two traffic types. Further, to satisfy the delay constraint of voice traffic, data rounds can be restricted to a maximum length, $L_{d_{max}}$ while the length voice rounds is determined only by the number of voice stations, N_v, and the length of voice packets.

We now examine the effects of varying the maximum data round length, $L_{d_{max}}$. Considering a deterministic system with perfect scheduling of arrivals, maximum throughput is achieved when the time between the start of successive voice rounds is exactly D_{max} [Fine & Tobagi 85]. Under these conditions, $N_v = N^{(0)}_{v_{max}}$. Further, every data round has length $L_d = L_{d_{max}}$. Thus, we have,

$$D_{max} = N_v(P_p + P_o + D_{max}V + 2t_{cd}C)/C + L_d + 2(2\tau_p + 2t_{cd})$$

$$\text{i.e., } N_v = \frac{C(D_{max} - L_d - 2(2\tau_p + 2t_{cd}))}{P_p + P_o + D_{max}V + 2t_{cd}C} \qquad (6.1)$$

where P_p and P_o are the preamble and overhead per packet respectively. The term $(2\tau_p + 2t_{cd})$ is the idle period between successive trains and appears twice because we are considering a cycle consisting of a voice train and a data train. Thus, $N_{v_{max}}$ decreases linearly as $L_{d_{max}}$ increases, reaching zero for some $L_{d_{max}} < D_{max}$.

With randomness introduced in the data arrival process and in the number of active voice stations due to silence suppression, L_d is less than $L_{d_{max}}$ and $N_{v_{max}}$ can be appreciable even with $L_{d_{max}} = D_{max}$. This is seen in the results of simulations of a 10 Mb/s, 1 km Expressnet with $D_{max} = 20$ ms and $G_d = 20\%$. With $L_{d_{max}} = 20$ ms, $N^{(1)}_{v_{max}}$ is 265 and remains at this value when $L_{d_{max}}$ is decreased to 10 ms. When $L_{d_{max}}$ is further decreased to 2 ms, $N^{(1)}_{v_{max}}$ increases to 285 stations and η_d decreases from 15% to 10% (Figure 6-2). As is to be expected from the throughput curves, data delay is approximately constant with $L_{d_{max}}$ in the range 10 to 20 ms, and increases by about a factor of 2 when $L_{d_{max}}$ is decreased to 2 ms. Thus, choosing $L_{d_{max}} < D_{max}/2$ yields a small increase in voice capacity at the expense of a large decrease in data throughput. Note that value of

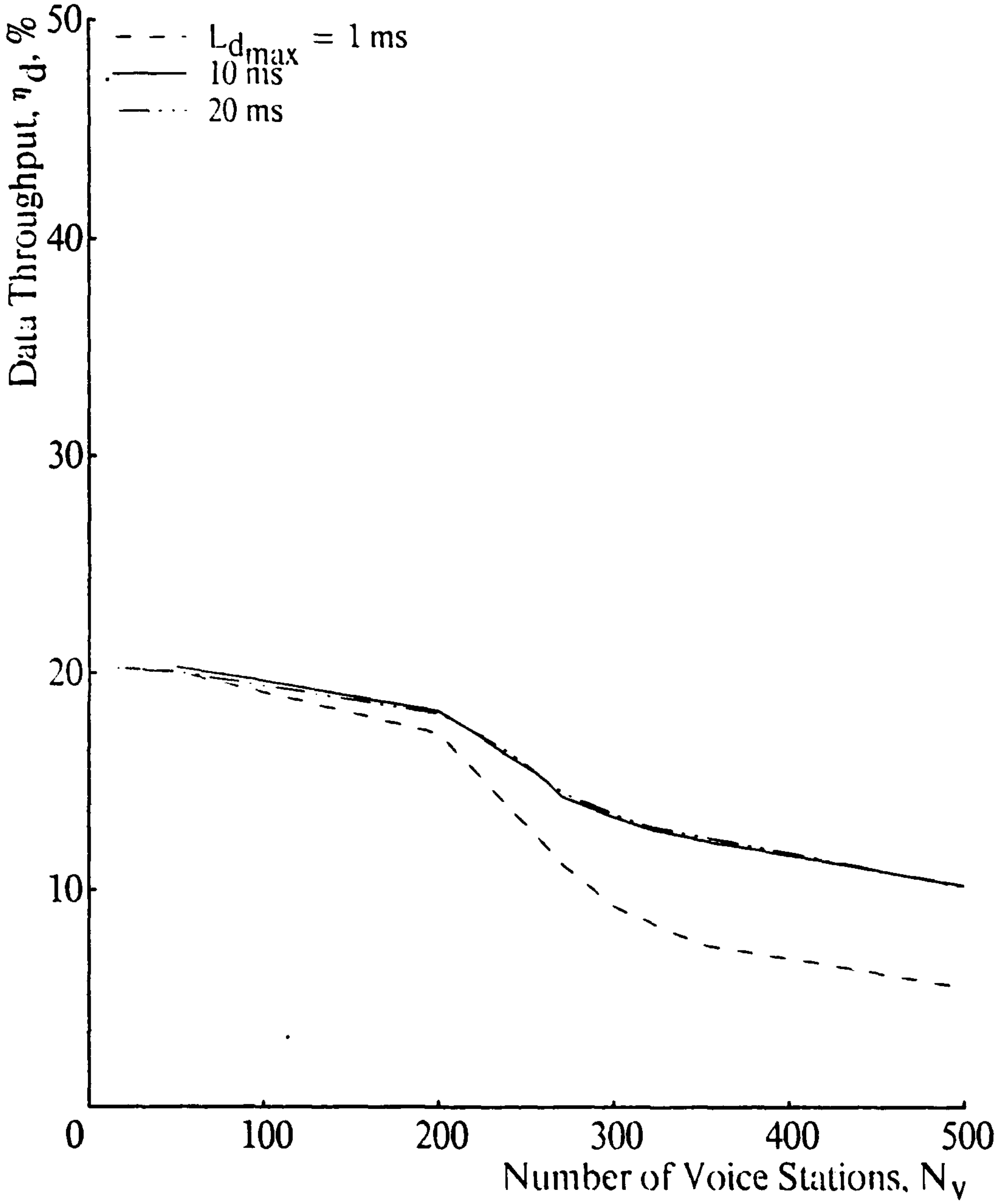


Figure 6-2: 10 Mb/s, 1 km Expressnet: Data Throughput vs. N_v D_{max} = 20 ms. G_d = 20%. Parameter, $L_{d_{max}}$.

$L_{d_{max}}$ at which η_d begins to drop rapidly is a function of both the number of data stations and D_{max}. For D_{max} = 200 ms, the knee point is below $D_{max}/2$, i.e., $L_{d_{max}} = D_{max}/2$ is a conservative choice while for D_{max} = 2 ms, the knee is at approximately $D_{max}/2$. We use $L_{d_{max}} = D_{max}/2$ in the rest of this Chapter.

6.2. Voice Traffic Performance

A system consisting of only voice stations operating without silence suppression is deterministic and the maximum number of voice stations that can be accommodated without loss, $N_{v_{max}}^{(0)}$, is given by Equation (6.1) with L_d set to the overhead per round, $2(\tau_p + t_{cd})$. For the parameter values summarized in Section 2.4 and Table 6-1, we compute $N_{v_{max}}^{(0)}$ for a 10 Mb/s, 1 km and a 100 Mb/s, 5 km Expressnet and several values of D_{max} (Table 6-2). It is evident that for the parameter ranges considered, inter-round propagation delay is insignificant compared to the per packet overhead. The latter is the cause of the reduced capacities at D_{max} = 2 and 20 ms.

Network Parameters:	
Max data round length, $L_{d_{max}}$	$0.5 D_{max}$
Max voice round length, $L_{v_{max}}$	Unlimited
Station Parameters:	
Packet overhead, P_o	10 bytes
Packet preamble, P_p	64 bits
Carrier detection time, t_{cd}	1.0 μs at C = 10 Mb/s 0.1 μs at C = 100 Mb/s
Voice Traffic Parameters:	
Minimum delay, D_{min}	0.125 ms at D_{max} = 2 ms 1.0 ms at D_{max} = 20, 200 ms

Table 6-1: Expressnet: Simulation Parameters

	D_{max}, ms	$N^{(0)}_{v_{max}}$
10 Mb/s, 1 km	2	67
($bC/Vc = 156$)	20	138
	200	154
100 Mb/s, 5 km	2	650
($bC/Vc = 1562$)	20	1378
	200	1541

Table 6-2: Expressnet: Voice capacity with only voice stations. Without silence suppression.

6.3. Minimum Voice Packet Length

We now examine the effect of varying P_{min} on Expressnet performance. As in the Token Bus, performance is only weakly dependant on P_{min} due to the round-robin nature of the protocol, provided that P_{min} is chosen small relative to P_{max}. Note that $P_{min} = VD_{min}$ and $P_{max} = VD_{max}$. If P_{min} is large small clips occur even for N_v well below $N_{v_{max}}$. Consider the situation when silence suppression is in effect, there is some data load, and $P_{min} = P_{max}/2$. Assume that the load is such that the cycle length, i.e., the time to the beginning of the next voice round, is $L = D_{max}/2$. Consider the left-most voice station, i (similar reasoning applies to any voice station). Assume that i transmits at the beginning of a voice round. Let the length of the next cycle be $L = D_{min} - \epsilon_1$, for some positive ϵ_1. At this point, station i will not yet be ready to transmit the next packet and so will lose its turn. Now, let the length of the succeeding cycle be $L = D_{min} + \epsilon_2$, for $\epsilon_2 > \epsilon_1$. Station i can now transmit. However, the time since its last transmission is $L - \epsilon_1 + L_2 + \epsilon_2 > D_{max}$. Hence, it suffers loss even though the average cycle length is much less than D_{max}. Note that ϵ_i varies with time owing to the variation in data load and the transition of voice stations between the talk and silent states.

Empirically, we have found that values of D_{min} on the order of $D_{max}/10$ or less minimize voice loss. Hence, in the subsequent evaluations, we use $D_{min} = 1$ ms for D_{max}

= 20 and 200 ms and D_{min} = 0.125 ms for D_{max} = 2 ms. The corresponding values of P_{min} are 8 and 1 bytes respectively.

6.4. Voice/Data Traffic

In this section we examine the performance of the Expressnet under various conditions with voice/data traffic. Network bandwidths of 10 and 100 Mb/s are considered with the folded-bus topology (Figure 6-1). Parameter values are summarized in section 2.4. Some parameter values specific to the Expressnet are listed in Table 6-1. In the following sections we discuss, in order, system, voice and data performance with integrated voice/data traffic.

6.4.1. System Measures

Owing to the round-robin nature of the scheduling discipline, the Expressnet is inherently fair as each active station can transmit 1 packet per round. Hence the only system performance metric that we consider is maximum throughput under various conditions.

Throughput

Maximum system throughput, η, is found to be a function primarily of channel length, d, and maximum voice delay, D_{max}. Variation in η is on the order of 1% or less for each 10% variation in data offered load, G_d. Silence suppression also has little effect on η. Table 6-3 gives η at φ = 0.5%, with G_d fixed at 20%, for various values of C, d and D_{max}. φ is chosen to be 0.5% because this is well below the value of 1% that we use as the limit of acceptability. Hence, the throughputs shown are lower bounds for voice/data systems with the range of parameters considered.

Throughput is seen to increase with D_{max}. Under heavy traffic conditions, voice packets are of length $P_{max} = D_{max}V$. Thus, as D_{max} is increased, the fixed overhead per packet is amortized over a larger number of voice samples. Throughput decreases with increase in d due to the effect of increased propagation delay, i.e., higher a. For D_{max} of 20 and 200 ms, η is almost constant with increase in C from 10 to 100 Mb/s. At φ = 0.5%,

	C = 10 Mb/s		C = 100 Mb/s
d =	1 km	5 km	5 km
D_{max} = 2 ms			
without suppression	40.7%	39.9%	45.8%
with suppression	39.6%	39.3%	43.0%
D_{max} = 20 ms			
without suppression	86.0%	86.0%	86.5%
with suppression	82.7%	83.2%	81.6%
D_{max} = 200 ms			
without suppression	98.2%	98.0%	98.2%
with suppression	96.5%	96.2%	92.6%

Table 6-3: Expressnet: Total System Throughput at $\varphi = 10\%$
$G_d = 20\%$

the sum of the mean voice and data round lengths are about D_{max}, independent of C. Thus η is a function of packet overhead and τ_p but not of C. However, with $D_{max} = 2$ ms, η *increases* with increase in C. This seemingly anomalous behaviour is caused by the limited length of data rounds, $L_{d_{max}} = D_{max}/2 = 1$ ms. For simplicity, we ignore interactive data packets in this explanation. At 10 Mb/s, the transmission time for a 1000 byte packet is 0.8 ms. Thus the propagation delay overhead of $2\tau_p = 50$ μs is incurred once per 0.8 ms. At 100 Mb/s, T_p decreases to 0.08 ms while the propagation delay remains the same. Thus, up to 12 packets may be transmitted in 1 ms and the propagation delay overhead is incurred once per $0.08\times12=0.96$ ms leading to higher efficiency.

Under the most stringent delay constraint used, $D_{max} = 2$ ms, the throughput is still appreciable at about 40% of C. At larger values of D_{max}, the protocol performs very well. Note that while there is a substantial increase in η achieved by increasing D_{max} from 2 to 20 ms, the increase owing to further increasing D_{max} to 200 ms is small.

6.4.2. Voice Measures

Next we consider the performance achieved by voice traffic and the effect on it of varying parameters such as data load. For the most part, the discussion deals with a 100 Mb/s, 5 km network and applies to lower bandwidth networks except where otherwise noted.

Loss

Figure 6-3 shows voice loss, φ, as a function of N_v with G_d fixed at 20% and D_{max} taking on values of 2, 20 and 200 ms. For each value of D_{max}, the performance is shown with and without silence suppression. Similarly to the Token Bus (Figure 5-4), as N_v increases from zero, there is no loss until $N_{v_{max}}^{(0)}$, which value is dependant on D_{max}. Thereafter, loss occurs and increases linearly with N_v. The knee is sharper without silence suppression than with. In the former case, variation in offered load occurs only due to variation in data traffic, while in the latter case, the number of active voice stations varies with time leading to the difference in knee shapes. After the knee, the slope of the curves is lower in the case of silence suppression because each additional station contributes a lower additional load.

This graph brings out two important factors. Firstly, increasing D_{max} from 2 ms to 20 ms causes a large increase of about 300% in system capacity (Table 6-4). However, the increase in capacity achieved by increasing D_{max} by another order of magnitude to 200 ms is much lower, about 30%. Thus, there is not much advantage in operating an Expressnet with D_{max} much larger than 20 ms. This corresponds well with the requirements for both intra-LAN traffic and traffic over a public telephone network. Secondly, the increase in capacity due to use of silence suppression is close to 2.5, the ratio of the average talkspurt length to the average silence length[16].

To study the effects of varying data loads on voice performance, we plot in Figure 6-4 φ versus N_v for several values of G_d. D_{max} is fixed at 20 ms. The shapes of the curves for different G_d are very similar, the main difference being in the point at which the knee

[16] Similar observations are made in [Fine 85]

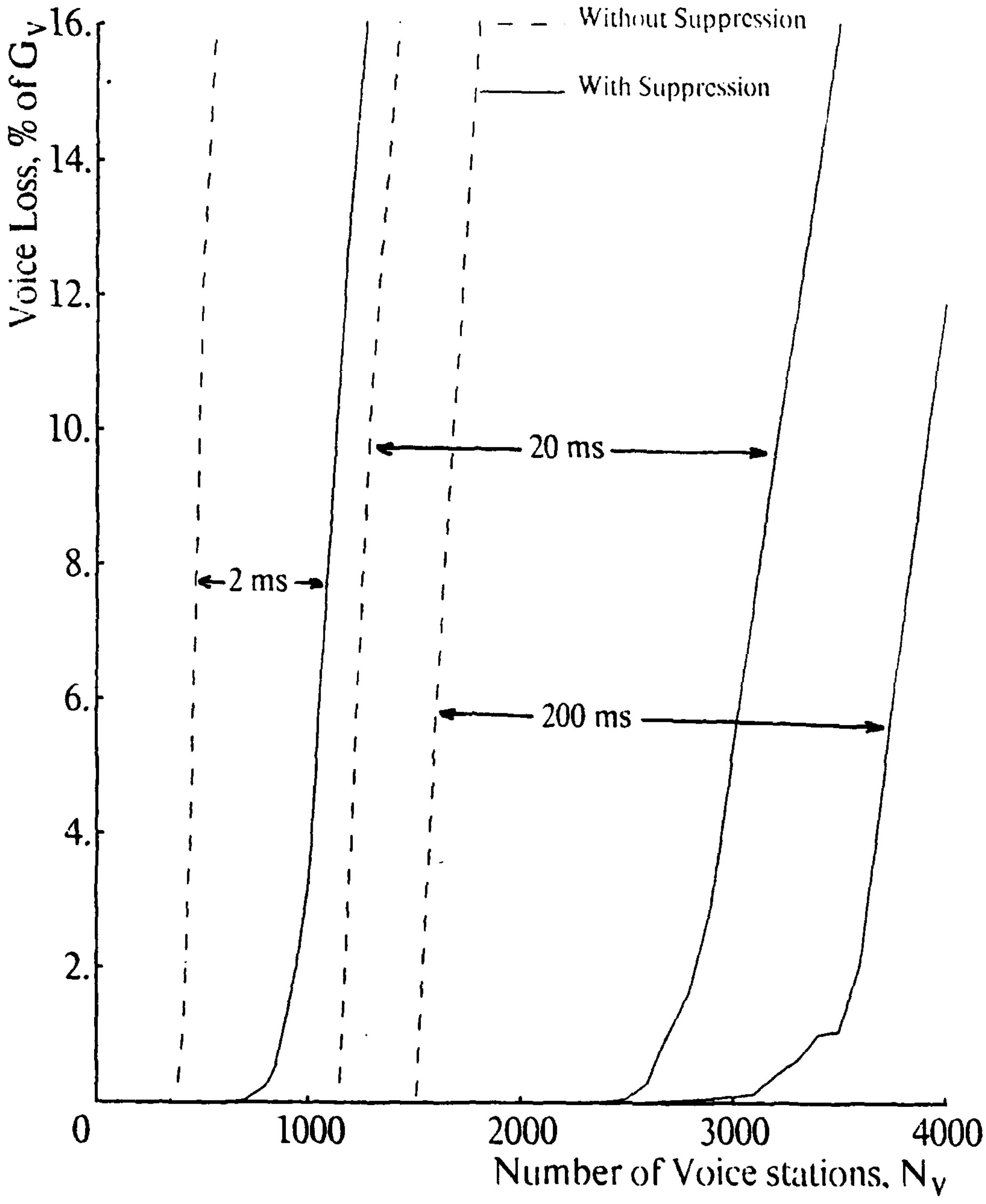


Figure 6-3: 100 Mb/s, 5 km Expressnet: Loss vs. N_v.
G_d = 20%. Parameter D_{max}.

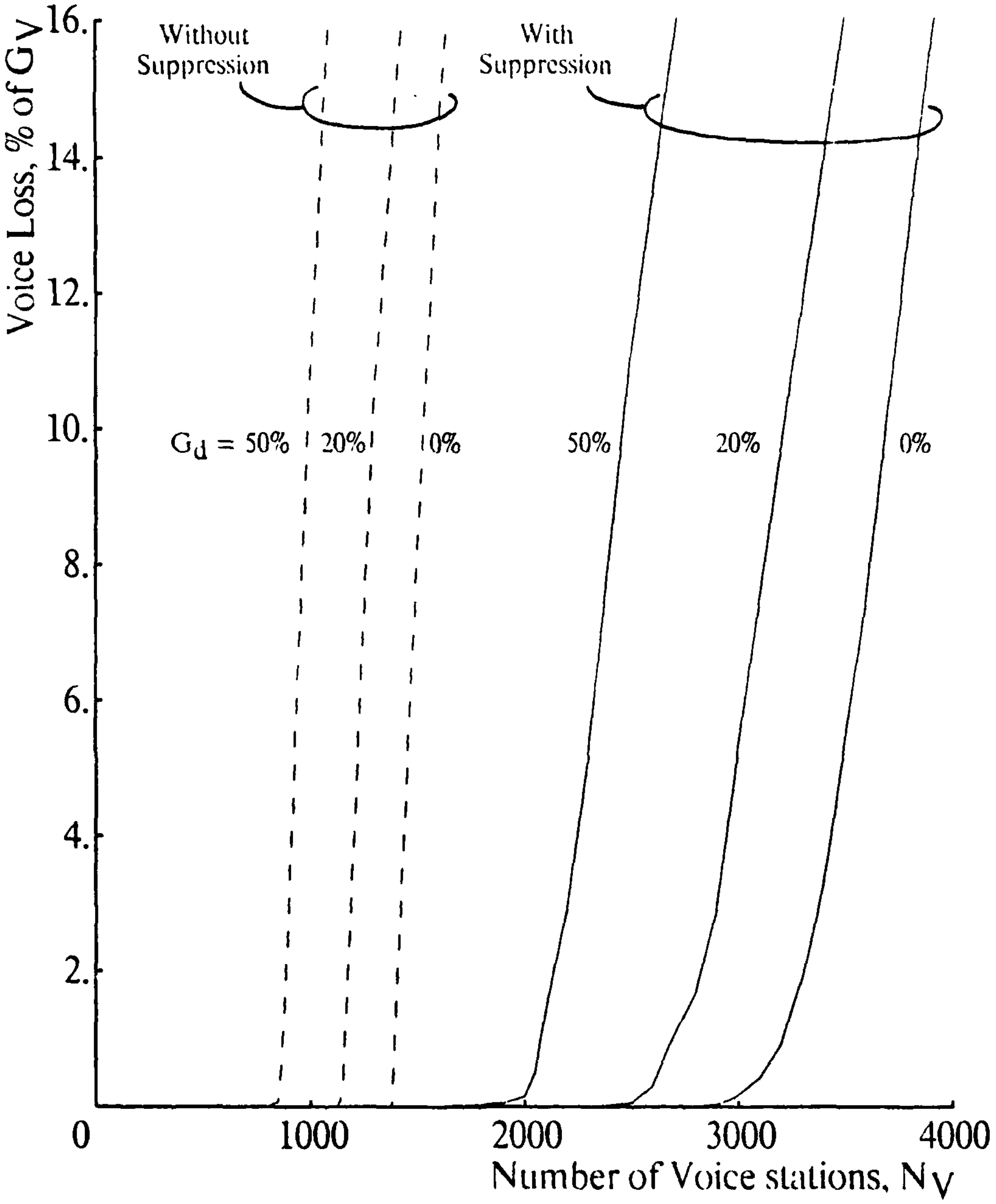


Figure 6-4: 100 Mb/s, 5 km Expressnet: Loss vs. N_v
D_{max} = 20 ms. Parameter G_d

	10 Mb/s, 1 km			100 Mb/s, 5 km		
Φ =	1%	2%	5%	1%	2%	5%
D_{max} = 2 ms						
without suppression	32	34	38	408	422	452
with suppression	77	82	95	880	944	1035
D_{max} = 20 ms						
without suppression	114	116	120	1159	1173	1215
with suppression	264	272	287	2694	2825	2990
D_{max} = 200 ms						
without suppression	153	155	160	1525	1541	1590
with suppression	355	363	380	3396	3595	3716

Table 6-4: Expressnet: Voice Capacity at φ = 1, 2, 5%.
10 Mb/s, 1 km and 100 Mb/s, 5 km. G_d = 20%.

occurs. The change in total throughput, $\eta = \eta_d + \eta_v$, is much less than G_d for the range shown (Table 6-5). Further, η_d is less than G_d due to the loss of data packet arrivals when the buffer in a data station is occupied by a packet awaiting transmission. Thus, the increase in voice throughput, η_v, due to a decrease in G_d is smaller than the change in G_d.

G_d	Without silence suppression			With silence suppression		
%	η_v	η_d	η	η_v	η_d	η
0	88.8	0.0	88.8	80.5	0.0	80.5
20	73.2	13.3	86.5	67.6	14.8	82.4
50	54.0	31.7	85.7	52.1	34.3	86.4

Table 6-5: 100 Mb/s, 5 km Expressnet: Throughputs at φ = 1%.
D_{max} = 20 ms.

We now consider the nature of the loss. Tables 6-6 and 6-7 show details on clipping when the total loss is 1% and 5% respectively. G_d is fixed at 20%. Clip lengths are higher with silence suppression than without due to the higher instantaneous fluctuations in

	$N_{v_{max}}$	Loss Length, T_l (ms)			Inter-loss time, T_{ilt} (s)	
		Mean	Std dev	Max	Mean	Std dev
D_{max} = 2 ms						
without suppression	408	0.12	0.10	0.5	0.012	0.010
with suppression	880	0.17	0.15	0.9	0.016	0.083
D_{max} = 20 ms						
without suppression	1159	0.24	0.16	0.7	0.030	0.013
with suppression	2694	0.65	0.67	11.8	0.041	0.143
D_{max} = 200 ms						
without suppression	1525	2.11	0.12	2.2	0.231	0.272
with suppression	3396	25.98	24.04	257.8	0.246	0.170

Table 6-6: 100 Mb/s. 5 km Expressnet: Clipping Statistics at $\varphi = 1\%$ $G_d = 20\%$. Parameter D_{max}.

	$N_{v_{max}}$	Loss Length, T_l (ms)			Inter-loss time, T_{ilt} (s)	
		Mean	Std dev	Max	Mean	Std dev
D_{max} = 2 ms						
without suppression	452	0.18	0.12	0.8	0.004	0.003
with suppression	1035	0.29	0.22	1.1	0.007	0.030
D_{max} = 20 ms						
without suppression	1215	1.06	0.22	1.6	0.021	0.000
with suppression	2990	1.43	1.00	13.9	0.029	0.079
D_{max} = 200 ms						
without suppression	1590	10.53	0.49	10.7	0.220	0.092
with suppression	3716	22.69	27.16	293.5	0.369	0.398

Table 6-7: 100 Mb/s, 5 km Expressnet: Clipping Statistics at $\varphi = 5\%$ $G_d = 20\%$. Parameter D_{max}.

traffic. Also, clip length increases with D_{max} due to the round-robin service. Loss occurs when the length of the voice plus data rounds exceeds D_{max}. In such cases, each voice station transmits a packet of length P_{max} once per round. Thus each station incurs loss once per round. Since the total loss is constant at 1% or 5%, the average clip length must increase as the time between clips increases. In all cases, except D_{max} = 200 ms with silence suppression, the mean clip length is on the order of 1 ms, with the maximum being less than 12 ms. These values are much lower than the 50 ms threshold cited by Campanella (see Section 2.3.2). Thus, loss would appear to the user as distortion rather than perceptible gaps. This is the preferred mode of loss for voice transport systems. In the exception, D_{max} = 200 ms with silence suppression, the mean is still less than the 50 ms threshold. However, the variance is high with the longest clip being about 260 and 290 ms at φ = 1 and 5% respectively. This situation is less desirable. The statistics at φ = 5% are similar, except that the mean clip lengths are higher due to the higher total loss.

Delay

The variation of D_v with N_v is shown in Figure 6-5 for several values of D_{max}. Curves are plotted for performance with the use of silence suppression as well as without. The network bandwidth and length are 100 M/bs and 5 km respectively.

Considering the curves without silence suppression, we see that as N_v is increased from zero upwards, delay increases approximately linearly from a small initial value up to D_{max}. The initial value is equal to the length of the data round, i.e., 0.125 ms when D_{max} = 2 ms and 1 ms when D_{max} = 20 and 200 ms. When delay reaches D_{max}, the combined lengths of the voice and data rounds is equal to D_{max}. Thereafter, all voice packets have length P_{max} and consequently delay D_{max}, the length of the voice round is directly proportional to N_v, and loss occurs. While delay is less than D_{max}, standard deviation of delay is small (Figure 6-6). When delay reaches D_{max}, standard deviation reaches a peak that increases with D_{max} and then drops abruptly to a very low value.

With silence suppression, the effects are similar except that the increase in delay with N_v is slower since the traffic presented by one additional voice station is 40% that of an additional voice station without silence suppression. The point at which delay reaches D_{max} is less well defined because of the greater variation in traffic as voice stations move between talk and silent states. This variation is in addition to the fluctuations in data

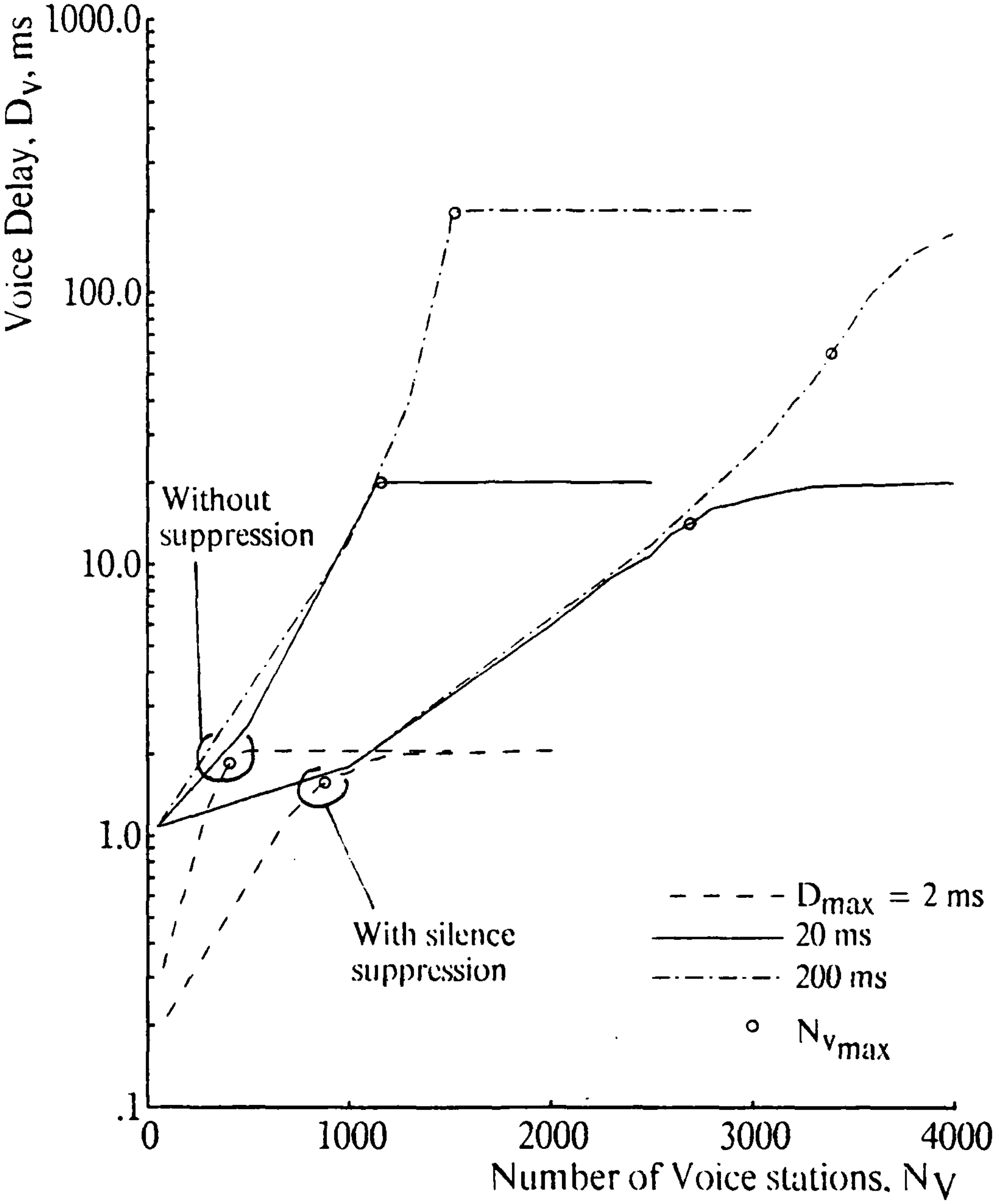


Figure 6-5: 100 Mb/s, 5 km Expressnet: Voice Delay, D_v vs. N_v
G_d = 20%. Parameter D_{max}.

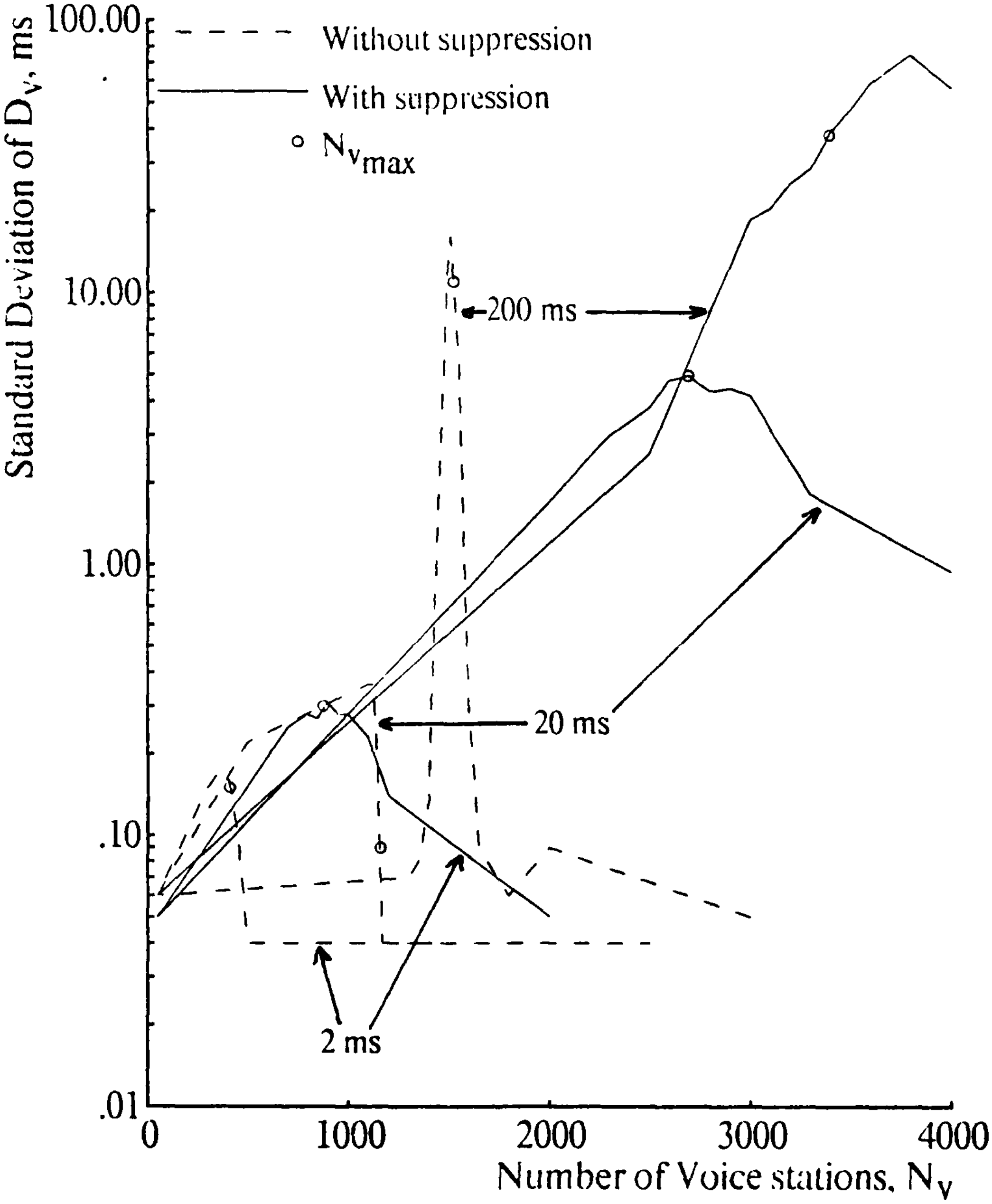


Figure 6-6: 100 Mb/s, 5 km Expressnet: Standard Deviation of D_v vs. N_v
G_d = 20%. Parameter D_{max}.

traffic, which are present with and without silence suppression. Standard deviation is higher with silence suppression than without and the decrease as N_v exceeds $N_{v_{max}}^{(0)}$ is more gradual due to the variation in voice traffic described above. In the worst case shown, standard deviation is less than half the mean, and is considerably lower in most cases. This may be attributed to the round robin scheduling of the Expressnet.

The only effect of increasing the average data offered load, G_d, is to shift the delay curves to the left (Figure 6-7). Standard deviation curves for $G_d = 0$ and 50% are similar in shape and peak value to the curve for $G_d = 20\%$, $D_{max} = 20$ ms in Figure 6-6, differing only in being offset to the left for larger G_d, and to the right for smaller G_d.

6.4.3. Data Measures

Next we consider the performance achieved by data traffic and the effects of various parameters on it. To recapitulate, data traffic consists of a mix of 50 byte and 1000 byte packets, representing interactive and bulk traffic respectively. Both types of data traffic may be transmitted only within the data rounds which alternate with voice rounds. The length of each data round is limited to half of the maximum acceptable voice delay, i.e., $D_{max}/2$. In the following, data throughput, η_d, refers to the total throughput of bulk and interactive traffic. Delay is measured separately for the two traffic types and is found to be similar in both mean and variance. Hence, only delay for interactive traffic is presented. We present results for a 100 Mb/s, 5 km network.

Throughput

For $G_d = 20\%$, the variation of η_d with N_v is shown in Figure 6-8 for various values of D_{max}, with and without silence suppression. Consider first the curves without silence suppression. For $N_v < N_{v_{max}}^{(0)}$, i.e., before the system capacity is reached, η_d decreases slightly with increase in N_v. As N_v increases beyond $N_{v_{max}}^{(0)}$, η_d decreases more rapidly owing to the limited data round length. For a given N_v, η_d decreases with increase in D_{max} because the voice round length increases with N_v (proportionately, for N_v greater than voice capacity) and each of the fixed number of data stations can transmit at most one packet in each 2-round cycle. This is seen clearly in the case of $D_{max} = 200$ ms. Thus if

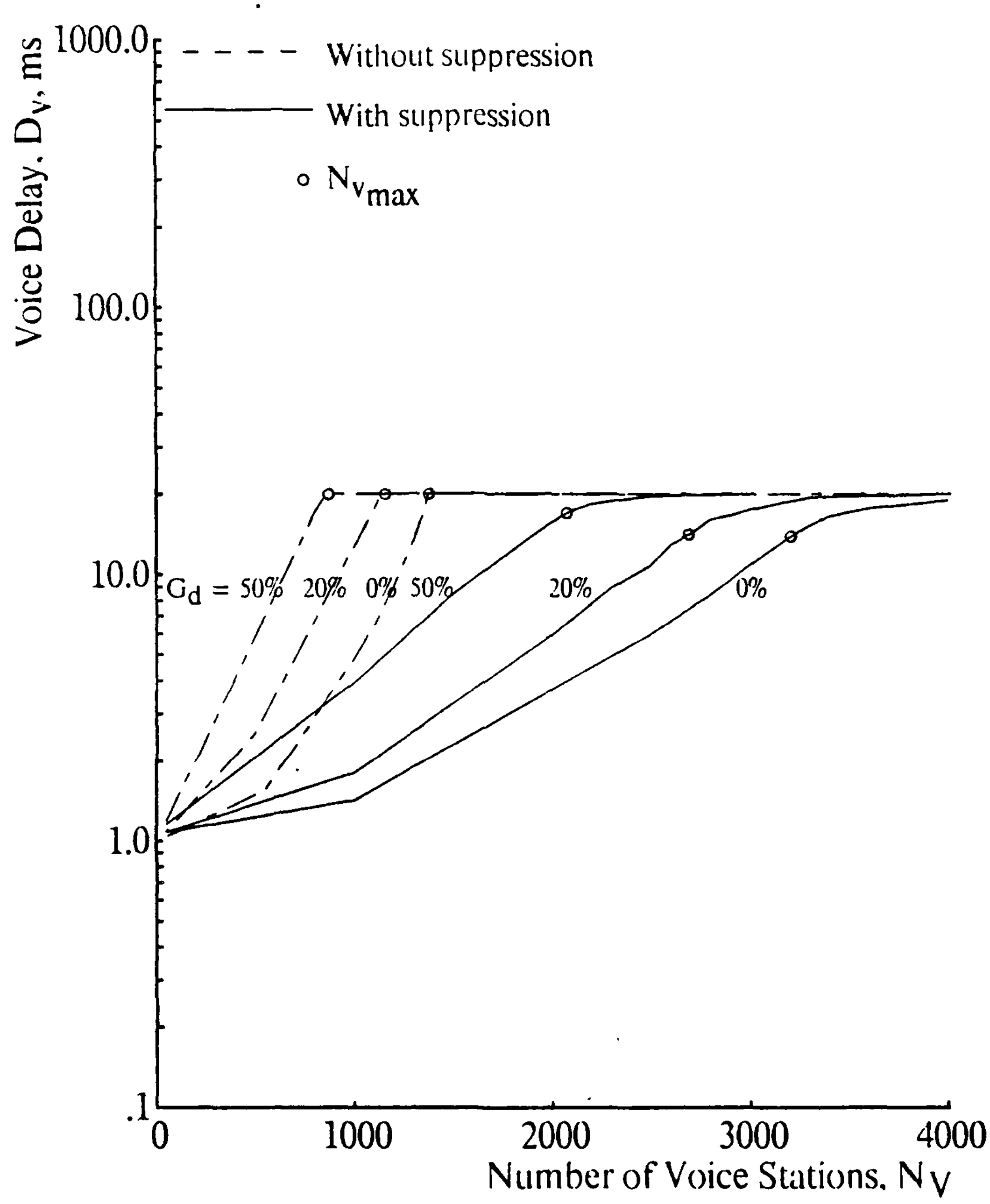


Figure 6-7: 100 Mb/s, 5 km Expressnet: Voice Delay, D_v vs. N_v
D_{max} = 20 ms. Parameter G_d

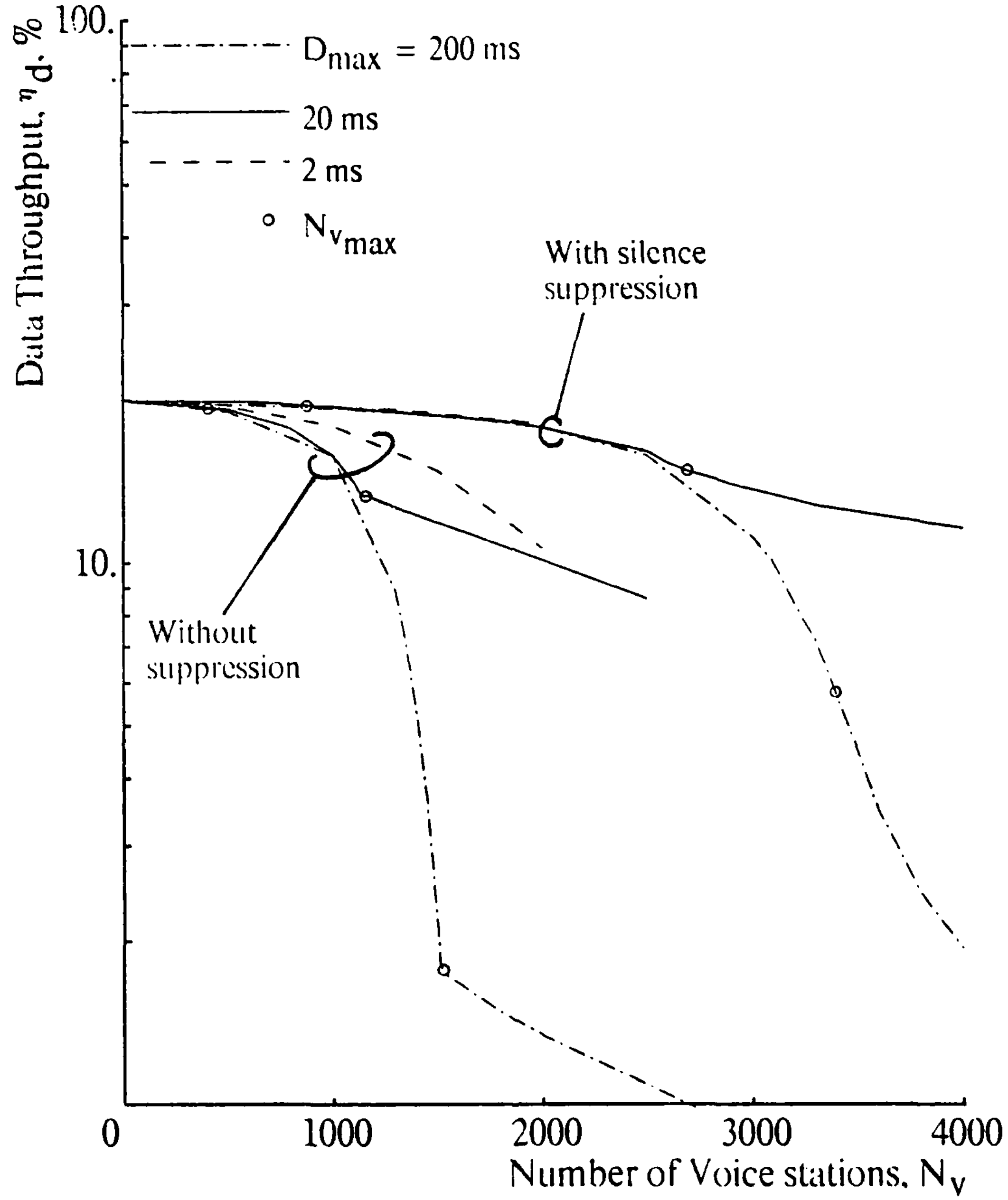


Figure 6-8: 100 Mb/s, 5 km Expressnet: Total Data Throughput vs. N_v
G_d = 20%. Parameter D_{max}.

voice performance is a priority, the usual case, larger values of D_{max} are preferable. If, however, data performance is of concern, lower values of D_{max} would be preferable. This trade-off is demonstrated in Table 6-8 in which η_d is shown for various values of voice loss and D_{max}.

Φ_{max}	1%	5%	10%
D_{max} = 2 ms			
without suppression	19.3%	19.3%	19.4%
with suppression	19.5%	19.3%	19.3%
D_{max} = 20 ms			
without suppression	13.3%	13.1%	12.7%
with suppression	14.8%	13.7%	13.0%
D_{max} = 200 ms			
without suppression	1.8%	1.7%	1.6%
with suppression	5.8%	2.9%	2.2%

Table 6-8: 100 Mb/s, 5 km Expressnet: η_d at φ = 1, 5, 10%. G_d = 20%. Parameter D_{max}.

The effect of silence suppression is to reduce the decrease in η_d with N_v because each additional voice station contributed a lower additional load. For a given voice loss, η_d is larger with silence suppression than without (Table 6-8). The difference is marginal at D_{max} = 2 ms, and appreciable at D_{max} = 200 ms.

The effect on η_d of varying G_d is shown in Figure 6-9. Both with and without silence suppression, the primary effect of a change in G_d is a shift of the curves vertically by an amount equal to the change in G_d.

Delay

Since each data station has a single packet buffer, as has each voice station, variation of data delay with N_v is expected to be similar to that of voice delay. The main difference is that data delay continues to increase with N_v beyond $N_{v_{max}}^{(0)}$ because a data packet

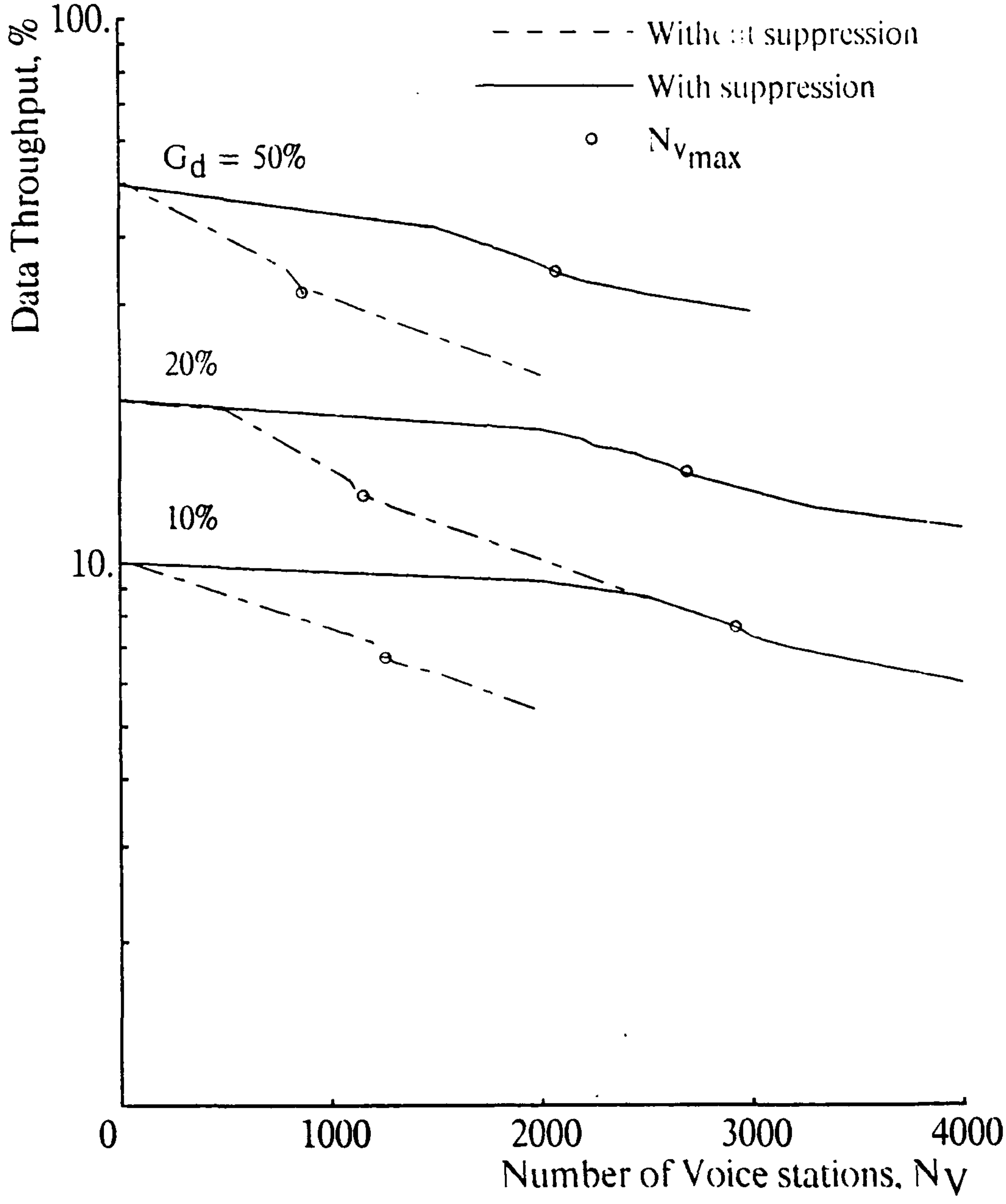


Figure 6-9: 100 Mb/s, 5 km Expressnet: Total Data Throughput vs. N_v D_{max} = 20 ms. Parameter G_d

remains in the buffer until it has been successfully transmitted, regardless of delay. This correspondence is seen by comparing the curve for data delay for any value of D_{max} in Figure 6-10 to that for voice delay for the same value of D_{max} in Figure 6-5. Varying G_d shifts the curves to the left for larger G_d and to the right for smaller G_d, similar to the effect in Figure 6-7.

Standard deviation, σ_d, is low, always less than the average (Figure 6-11). For small N_v, variations in data load cause the length of the data round, L_d, to fluctuate resulting in a corresponding fluctuation in the length of the voice round, L_v. Once N_v is sufficiently large, L_v is constant at approximately $N_v D_{max} V/C$, and $L_d = L_{d_{max}}$. Hence, σ_d reaches a plateau.

6.5. Summary

For the ranges of parameters considered, the Expressnet is found to have a total throughput strongly dependant on D_{max} but weakly dependant on bandwidth and on the fraction of traffic that is offered by data stations. Without silence suppression and with an offered data load of 20%, a 100 Mb/s network is shown to be able to support about 1200 active voice stations with a maximum delay of 20 ms and less than 1% loss. With the use of silence suppression, the capacity increases to nearly 3000 voice stations. Since during the busiest period of the day only 10-20% of voice stations in a system are active, the network could have 12,000 and 30,000 stations attached with and without silence suppression respectively. Note that a conversation requires two active stations. The loss is comprised of frequent short clips that are subjectively more tolerable than longer clips. There is a high correlation between the lengths of adjacent clips, indicating that every voice packet suffers similar clips. The effect of varying data load is to change voice capacity without affecting the quality of the voice service.

Data traffic is seen to achieve a fairly constant share of the network bandwidth in the preferred region of operation, i.e., when total offered load is less than C and voice loss is less than 1%. Under higher loads, data throughput falls as voice traffic gets precedence. Data delay increases with N_v to some value related to D_{max} and thereafter increases only

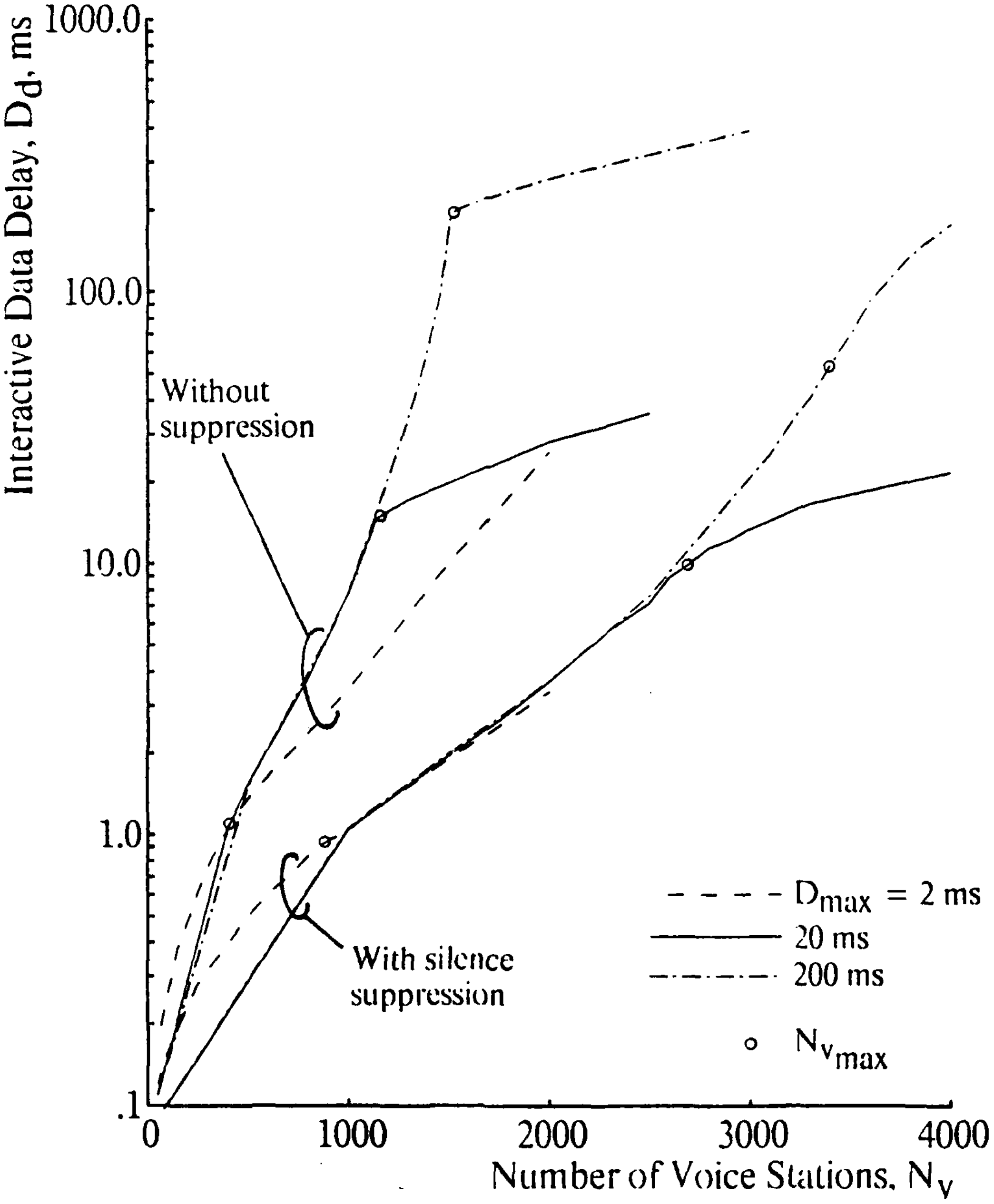


Figure 6-10: 100 Mb/s, 5 km Expressnet: Interactive Data Delay vs. N_v. G_d = 20%. Parameter D_{max}.

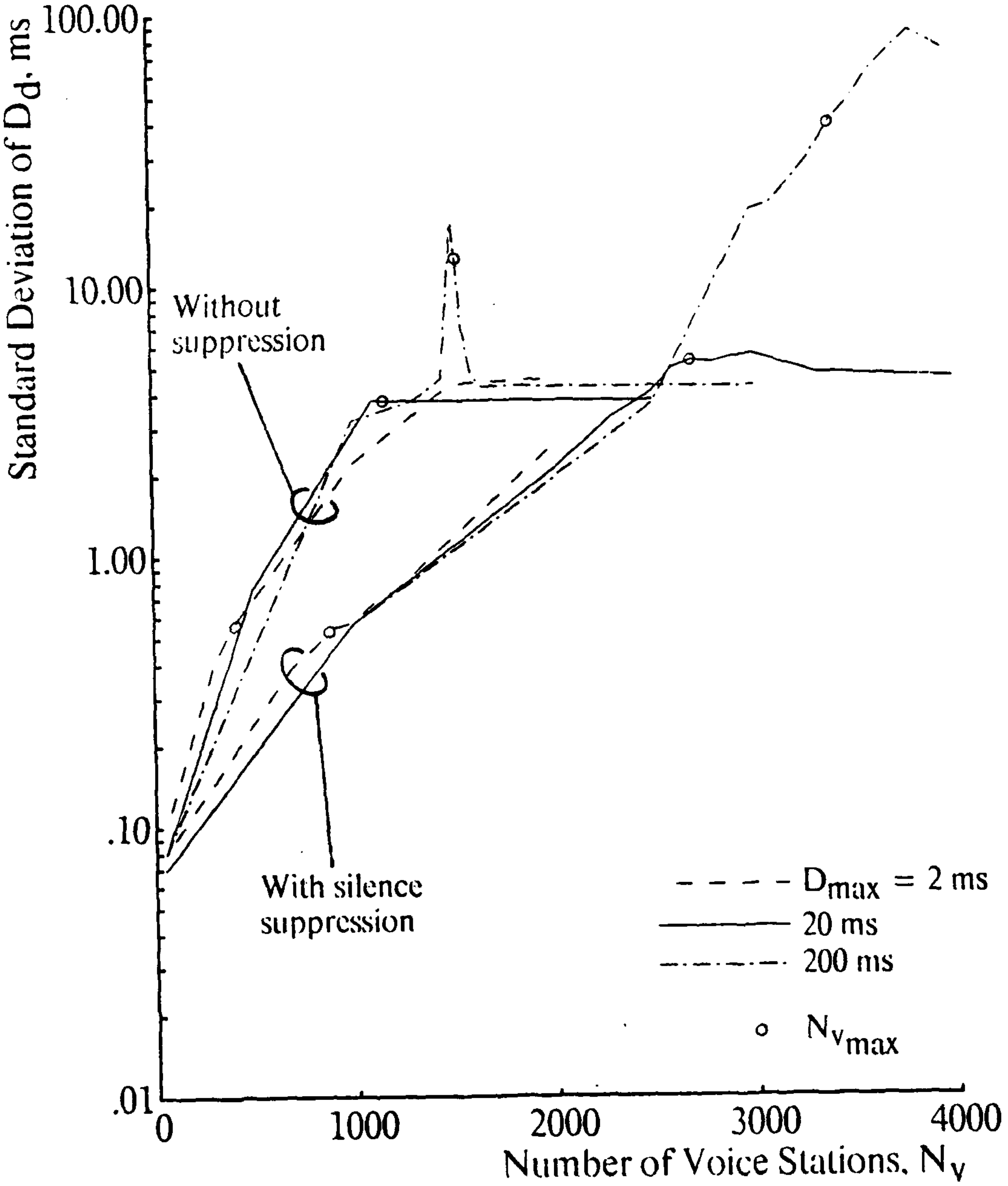


Figure 6-11: 100 Mb/s, 5 km Expressnet: Standard Deviation of D_i vs. N_v G_d = 20%. Parameter D_{max}.

gradually. Standard deviation of delay is low under low loads and stabilizes under overload.

Performance is shown to be relatively insensitive to the value of the voice protocol parameter, P_{min}, provided that P_{min} is chosen much less than P_{max}. If P_{min} is relatively large, small amounts of loss can occur even with N_v much lower than $N^{(1)}_{v_{max}}$. A study of the maximum data round length, $L_{d_{max}}$, indicates trade-offs between $N_{v_{max}}$ and η_d, with larger values of $L_{d_{max}}$ favouring data traffic at the expense of voice. This trade-off is less significant with silence suppression than without.

We note that our conclusions regarding the voice capacity are similar to those in an earlier study [Fine 85].[17] Our work extends Fine's study in several respects. With regard to voice performance, we investigate the nature of the clips in addition to the total clipping. With regard to data performance, Fine makes the heavy traffic assumption that all data rounds are of maximum length. In contrast, we model the performance of individual data stations under various loadings.

[17] Indeed, this adds to the credibility of the simulators used in our work and in the earlier work.

Chapter 7
Results: Comparative

Having examined the performance of the individual networks, we are in a position to present a comparative evaluation. As in the previous chapters, we first examine voice performance and then data performance in an integrated voice/data environment The emphasis is on contrasts between the networks, and on identifying the regions of good performance of each. Topics such as the optimization of P_{min}, and the effects of protocol-specific parameters such as TRT in the case of the Token Bus and $L_{d_{max}}$ in the Expressnet are covered in Chapters 4 - 6.

The protocols selected for evaluation span a wide range of available broadcast bus protocols. By restricting our attention to bus networks we are able to make uniform assumptions regarding parameters such as carrier sense time and synchronization preamble lengths which are dependant on the physical transmission medium. Due to the popularity of the broadcast bus topology, this class includes a large fraction of local area networks (see Chapter 1). The Ethernet is a contention-based protocol that has proved useful for data traffic and is coming into widespread use. The Token Bus and Expressnet are contention-free round-robin schemes of the DAMA class [Fine & Tobagi 84]. Depending on the selection of various parameters, several protocols in the DAMA class, such as FASNET and various ring protocols, have performance similar to that of the Token Bus or Expressnet and hence our results are indicative of the performance of such protocols also.

We compare the networks at bandwidths of 10 and 100 Mb/s. A summary of the simulation parameters is given in Table 7-1 and Section 2.5. As in the preceding chapters, we first present upper bounds for voice traffic only. This is followed by a detailed comp^arison of the protocols with integrated voice/data traffic.

Network Parameters:	
Token Bus:	
Max data token rotation time	$\geq D_{max}$
Max voice token rotation time	Unlimited
Expressnet:	
Max data round length, L_d	$D_{max}/2$
Max voice round length, $L_{v_{max}}^{max}$	Unlimited
Station Parameters:	
Packet overhead, P_o	10 bytes
Packet preamble, P_p	64 bits
Packet buffers	1
Carrier detection time, t_{cd}	1.0 μs at C = 10 Mb/s 0.1 μs at C = 100 Mb/s
Voice Parameters:	
Ethernet:	
Minimum delay, D_{min}	$0.4D_{max}$ at D_{max} = 2 ms $0.8D_{max}$ at D_{max} = 20, 200 ms
Token Bus, Expressnet:	
Minimum delay, D_{min}	0.125 ms at D_{max} = 2 ms 1.0 ms at D_{max} = 20, 200 ms

Table 7-1: Summary of simulation parameters

7.1. Voice Traffic Upper Bounds

We first present in Table 7-2 the maximum number of voice stations that each of the networks under consideration can support without loss. Two configurations are assumed: 10 Mb/s, 1 km and 100 Mb/s, 5km. Silence suppression is not used and D_{max} = 2, 20 and 200 ms. The values for the Ethernet are from our simulation while simple analytical formulae are used for the other networks. For the Token Bus, a random ordering of the stations in the logical ring is assumed with the mean distance between successive stations in the ring being $\tau_p/3$. $N_{v_{max}}$ is given in Equation (5.1) (page 111) for the Token Bus and in Equation (6.1) (page 125) for the Expressnet. The theoretical maximum is given by $\lfloor C/V \rfloor$.

	D_{max}	Ethernet*	Token Bus**	Expressnet	Theor Max
10 Mb/s,	2 ms	30	51	67	156
1 km	20 ms	100	129	138	156
	200 ms	125	153	154	156
100 Mb/s,	2 ms	-	165	650	1562
5 km	20 ms	125	848	1378	1562
	200 ms	1100	1441	1541	1562

* from simulation.

** random ordering of stations in the logical ring.

Table 7-2: System Voice Capacity, $N_v^{(o)}$
C = 10, 100 Mb/s. Voice stations only, without silence suppression.

Considering the 10 Mb/s case, we see that the Ethernet voice capacity is a small fraction, about 20%, of the theoretical maximum at D_{max} = 2 ms. The capacity of the other networks is substantially larger. In the Ethernet, in addition to the packet overhead, there is inefficiency due to collisions which increases as the packet length decreases. When D_{max} is increased to 20 and 200 ms, the capacity of the Ethernet increases to acceptable fractions of the network bandwidth due to a reduction in collisions. The other networks also show increases, though of a smaller magnitude.

For a 100 Mb/s, 5 km network, parameter $a=\tau_p/T_p$ increases by a factor of 50 compared to the 10 Mb/s, 1 km network. Thus, the Ethernet capacity is a small fraction of network bandwidth even at larger values of D_{max}. For the Token Bus, the effect of propagation delay in passing the token, $\tau_p/3$ on the average due to the assumption of random ordering, is evident in that the capacity increase is less than a factor of 10 compared to the corresponding capacities at 10 Mb/s. This is more pronounced at lower values of D_{max}. For the Expressnet, since the overhead of τ_p is incurred only once per round rather than once per packet, the increase in a has no effect on throughput. We note that under the optimum ordering the performance of the Token Bus is very similar to that of the Expressnet.

7.2. Voice/Data Traffic

From the preceding section, it is clear that the Ethernet is not a viable option for voice transmission at 100 Mb/s under the conditions considered. The Token Bus and Expressnet are both seen to merit further study at the higher speed of 100 Mb/s. In the remainder of this chapter, we compare the performance of the various networks in further detail, first at 10 Mb/s and then at 100 Mb/s for each of the performance measures. The Ethernet will be included only in comparisons of performance at 10 Mb/s.

7.2.1. Voice Measures

We examine first the performance of voice traffic and the effects of data traffic on it. The key measure of voice performance is loss. Delay is of importance primarily from a system design point of view.

Loss

We consider first the influence of maximum voice delay, D_{max}, and then that of data offered load, G_d, on voice loss. For a 10 Mb/s, 1 km network, the variation of φ with N_v is shown in Figure 7-1 for the Ethernet, Token Bus (with TRT $= D_{max}$) and Expressnet. $D_{max} = 20$ ms, $G_d = 20\%$, and the curves with and without silence suppression are shown. The shapes of the curves are similar for the different networks. For low values of N_v, there is no loss. At the point at which loss begins, there is a well-defined knee. Above the knee, loss increases rapidly. The knee is less well-defined in the case of the Ethernet. This may be attributed to the random nature of the access protocol compared to the more orderly round-robin schemes used in the Token Bus and Expressnet.

In the case without silence suppression, the curves of the Token Bus and Expressnet are almost identical. This is to be expected since the scheduling mechanisms are similar and propagation delay, which is the major difference between the two protocols, is relatively small. When silence suppression is used, however, the Expressnet performs better than the Token Bus. This is due to the assumption that voice stations in the silent state remain in the logical ring in the Token Bus to avoid the overhead associated with leaving and reentering the ring. Thus, each silent voice station transmits one packet consisting of $P_{ov} = 10$ bytes of overhead each round to pass the token.

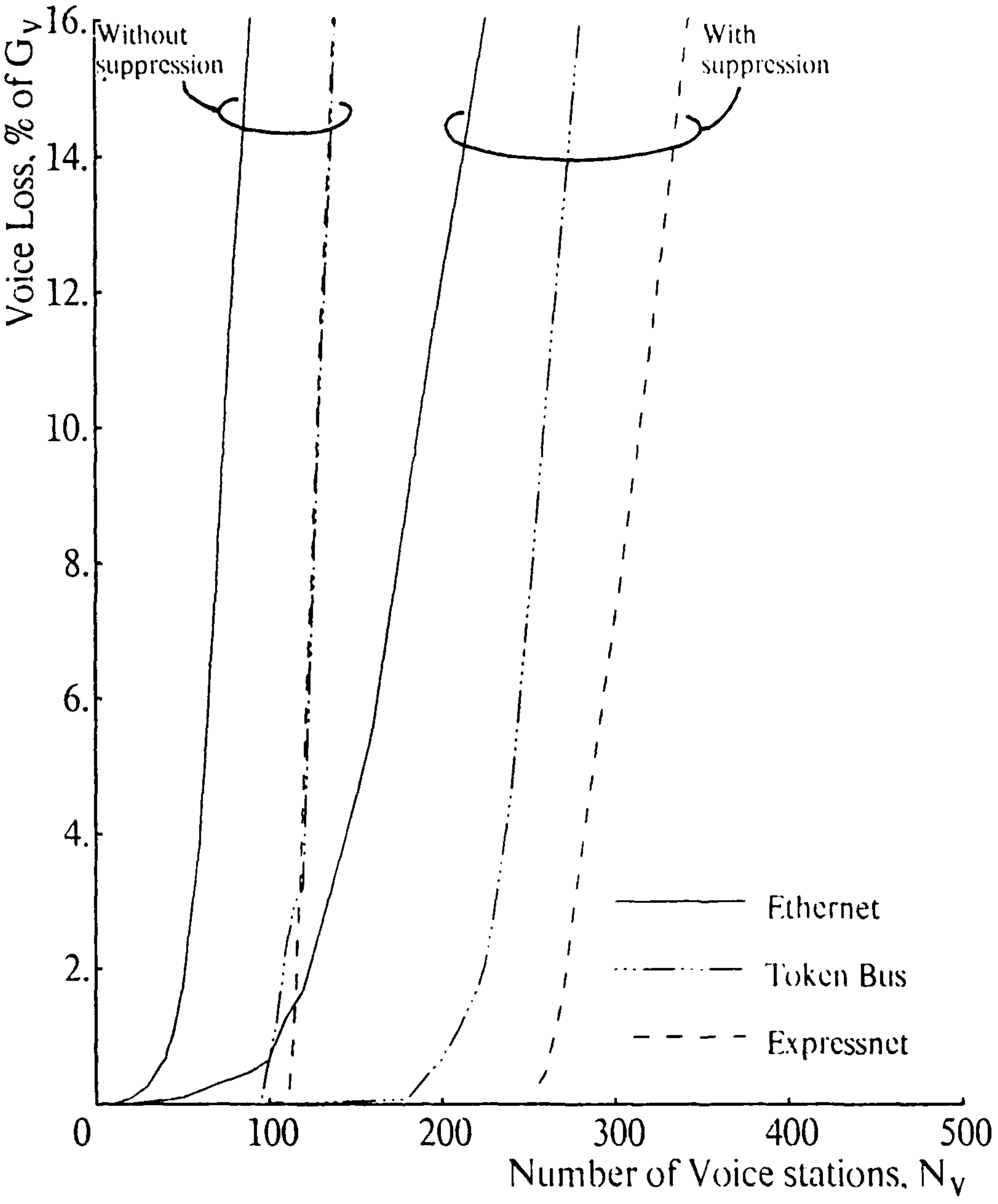


Figure 7-1: 10 Mb/s, 1 km: Loss vs. N_v
Ethernet, Token Bus, Expressnet
$G_d = 20\ \%$, $D_{max} = 20$ ms.

	D_{max}, ms		
	2	20	200
Without silence suppression:			
Ethernet	1	43	117
Token Bus	~0	103	152
Expressnet	32	114	152
With silence suppression:			
Ethernet	2	103	259
Token Bus	11	207	339
Expressnet	77	265	363

Table 7-3: 10 Mb/s, 1 km: Voice Capacity at $\varphi = 1\%$, $D_{max} = 2, 20, 200$ ms
Ethernet, Token bus, Expressnet
$G_d = 20\%$, with and without silence suppression

A summary of the loss curves is given in Table 7-3 in which the voice capacity, $N^{(1)}_{v_{max}}$, is shown for D_{max} = 2, 20 and 200 ms. Considering first the Ethernet, when D_{max} is increased from 2 to 20 ms, $N_{v_{max}}$ increases from close to zero to about 1/3rd of the maximum capacity. When D_{max} is increased from 20 to 200 ms, $N_{v_{max}}$ more than doubles. At low values of D_{max}, i.e., short packets, the Ethernet suffers both from relatively higher overhead per packet and from increased collision rates. As D_{max} increases, both factors decrease leading to the large increase in $N_{v_{max}}$. In the case of the Token Bus and Expressnet, there is a large increase when D_{max} is increased from 2 to 20 ms. However, when D_{max} is increased from 20 to 200 ms, the increase in capacity is lower. In these networks, inefficiency arises primarily due to packet overhead. In the Token Bus, at small D_{max}, propagation delay becomes significant and hence capacity is lower than in the case of the Expressnet.

To quantify the effect of varying data loads on voice performance, we present in Table 7-4 $N^{(1)}_{v_{max}}$ for G_d = 0, 20 and 50% with D_{max} fixed at 20 ms with silence suppression in

	Data Offered Load, G_d		
	0%	20%	50%
Ethernet	257	103	0
Token Bus (TRT = 20 ms)	232	207	197
(TRT = 25 ms)	232	187	153
Expressnet	291	265	158

Table 7-4: 10 Mb/s, 1 km: Voice Capacity at φ = 1%, G_d = 0, 20, 50%.
Ethernet, Token bus, Expressnet.
D_{max} = 20 ms, with silence suppression.

effect. Without silence suppression, the picture is similar except that the figures are lower by a factor of 2-2.5. The figures for the Ethernet show clearly the effect of the lack of priority for voice traffic: as G_d increases from 0 to 50%, $N^{(1)}_{v_{max}}$ decreases from 257 to 0. In the Token Bus, with TRT = 20 ms chosen to give high priority to voice, voice performance is not very sensitive to data offered load. When N_v reaches a value such that the token rotation time exceeds TRT, data stations do not transmit at all according to the protocol (Section 5.1). The choice of $L_{d_{max}}$ = 10 ms in the Expressnet gives weaker priority to voice: data stations are not denied service completely (see Section 6.1.1). Hence, G_d has a greater effect on $N_{v_{max}}$ than in the case of the Token Bus, but smaller than in the case of the Ethernet. We note that the priority mechanisms in the round-robin protocols are not intrinsic and could be implemented in either protocol with similar effects.

These differences may be clarified by examining the total throughput, η, and its division between voice and data, η_v and η_d, at φ = 1% (Table 7-5). In the Ethernet, η_d = G_d. Owing to the random arrivals of data packets, voice performance suffers and loss reaches 1% at smaller values of η as G_d increases. Thus, η_v decreases by an amount larger than the increase in η_d. In the two round-robin protocols, however, η is not strongly dependant on G_d. Thus, the decrease in η_v with increase in G_d is approximately equal to the increase in η_d. We note that the choice of TRT = 25 ms in the Token Bus, yields η_d = 14.5% and 36.0% for G_d = 20% and 50% respectively, very close to the values in the case of the Expressnet.

G_d %	Ethernet			Token Bus (TRT = 20 ms)			Expressnet		
	η_v	η_d	η	η_v	η_d	η	η_v	η_d	η
0	65.4	0.0	65.4	59.2	0.0	59.2	75.1	0.0	75.1
20	25.9	20.0	45.9	53.2	10.3	63.5	68.4	14.9	83.3
50	0.0	50.0	50.0	50.7	12.7	63.4	50.6	35.5	86.1

Table 7-5: 10 Mb/s, 1 km: Throughputs at $\varphi = 1\%$ for $G_d = 0, 20, 50\%$.
Ethernet, Token bus, Expressnet.
$D_{max} = 20$ ms. With silence suppression

With $G_d = 0\%$ and silence suppression in effect, the capacity of the Ethernet, 257 stations, exceeds that of the Token Bus, 232 stations. This seemingly anomalous behaviour is due to the overhead caused by silent stations taking part in the token passing in the Token Bus.

Finally, we examine the nature of the loss in the different networks. Statistics on clip length and inter-clip time are shown in Table 7-6 with $\varphi = 1\%$, $D_{max} = 20$ ms and $G_d = 20\%$. We see that in the Ethernet, the mean clip length is less than 10 ms but the variance is high with the maximum being close to 200 ms. This can be more of an annoyance to the listener than the clips in the Token Bus and Expressnet which have small means and maximum values. In the Ethernet, the correlation of the length of adjacent clips is found to be close to zero. This occurs because of the random nature of the access protocol. While one packet from a station might suffer a number of collisions and hence large loss, the next might be successful on the first attempt. In the 2 round-robin protocols, there is significant correlation which increases to close to unity for large φ.

C = 100 Mb/s

We now turn our attention to a 100 Mb/s, 5 km network. Here, we study only the Token Bus and Expressnet. Qualitatively, voice performance is similar to the 10 Mb/s networks (Figure 7-2). The principal difference is in the magnitude of $N^{(1)}_{v_{max}}$ (Table 7-7). We see the effect of the propagation delay incurred in passing the token in the Token Bus

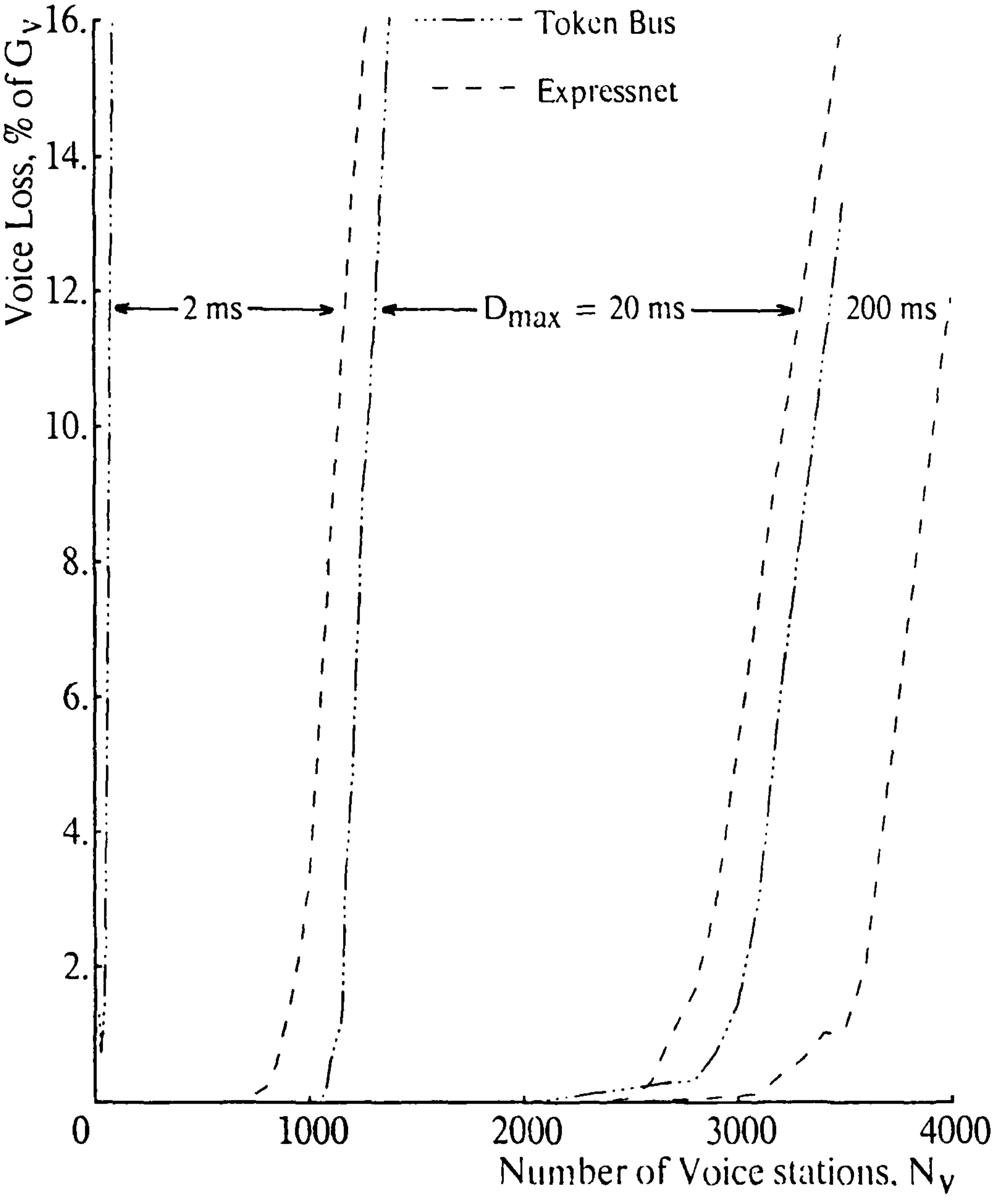


Figure 7-2: 100 Mb/s, 5 km: Loss vs. N_v
Token Bus, Expressnet
$G_d = 20$ %.

	$N_{v_{max}}$	Clip Length, T_l (ms)			Inter-clip time, T_{ilt} (s)	
		Mean	Std dev	Max	Mean	Std dev
Ethernet	103	7.5	14.7	176.1	0.75	0.747
Token Bus	207	0.4	0.24	2.8	0.04	0.048
Expressnet	265	1.1	1.18	17.9	0.11	0.314

Table 7-6: 10 Mb/s, 1 km: Clipping Statistics at $\varphi = 1\%$
Ethernet, Token bus, Expressnet
$D_{max} = 20$ ms, $G_d = 20\%$, with silence suppression

	D_{max}, ms		
	2	20	200
Without silence suppression:			
Token Bus	~0	757	1429
Expressnet	408	1159	1525
With silence suppression:			
Token Bus	15	1132	2936
Expressnet	880	2694	3396

Table 7-7: 100 Mb/s, 5 km: Voice Capacity at $\varphi = 1\%$, $D_{max} = 2, 20, 200$ ms.
Token bus, Expressnet.
$G_d = 20\%$. With and without silence suppression.

in the lower capacities compared to the Expressnet. This is especially evident at small D_{max} when the voice capacity of the Token Bus is close to zero. At $D_{max} = 20$ ms, the capacity increases to about 50% of the maximum, and to about 90% with $D_{max} = 200$ ms. With the Expressnet, even at 100 Mb/s and 5 km, propagation delay is not significant. Inefficiency is due almost entirely to packet overhead. We note that increasing TRT such that the data throughput is the same in both networks would decrease the voice capacity of the Token Bus by 5-15%.

Delay

Considering voice delay, we show in Figure 7-3 D_v as a function of N_v for $D_{max} = 20$ and 200 ms with silence suppression for the various networks. In the case of the Ethernet, delay is close to D_{max} even at small N_v owing to the choice of $P_{min} = 0.8D_{max}$ to maximize capacity (Section 4.5.2). Even under heavy traffic conditions the Ethernet access protocol allows some new packets to be transmitted with low delay while older packets are in the retransmission back-off phase after multiple collisions. Thus, even with $N_v \gg N_{v_{max}}$, D_v is less than D_{max}. The Token Bus and Expressnet exhibit similar delay characteristics with D_v increasing approximately linearly with N_v and reaching D_{max} at $N_v \approx N_{v_{max}}$. Note that for low N_v, D_v is lower in the case of the Expressnet than the Token Bus due to the assumption in the latter that idle voice stations continue to participate in the token passing. Variance of delay is low in the two round-robin schemes, and drops to close to zero when D_v reaches D_{max}. The Ethernet exhibits higher variance increasing monotonically with N_v.

7.2.2. Data Measures

Next we consider the performance achieved by data traffic and the effects of various parameters on it. To recapitulate, data traffic consists of a mix of 50 and 1000 byte packets, representing interactive and bulk traffic respectively. Throughput figures are the aggregate for both traffic types, while delay is computed separately. Since delay characteristics are similar for the two traffic types, we report only delay for interactive data.

Throughput

In Figure 7-4 total data throughput, η_d, is plotted as a function of N_v for $D_{max} = 20$ ms and $G_d = 20$ and 50%. The performance of the Ethernet and Expressnet are similar with η_d decreasing gradually with increasing N_v. Data packets receive lower priority in the Expressnet. However, the total throughput for a given N_v is higher in the case of the Expressnet and hence η_d is higher than for the Ethernet. In the Token Bus, with TRT = D_{max}, data throughput drops rapidly to zero as N_v approaches $N_{v_{max}}$, i.e., as the round length reaches D_{max}.

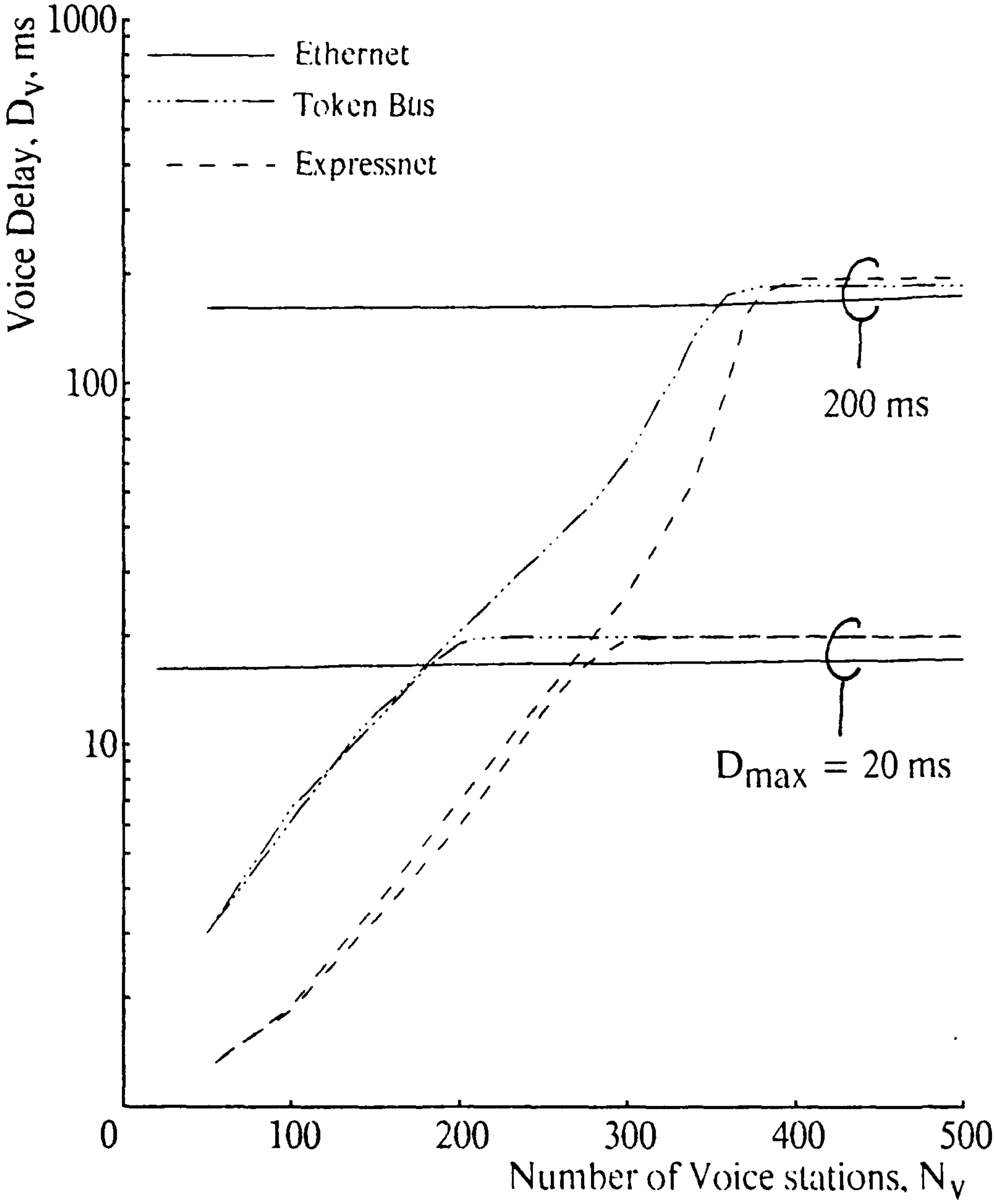


Figure 7-3: 10 Mb/s, 1 km: Voice delay vs. N_v
Ethernet, Token bus, Expressnet
$G_d = 20\%$, with silence suppression

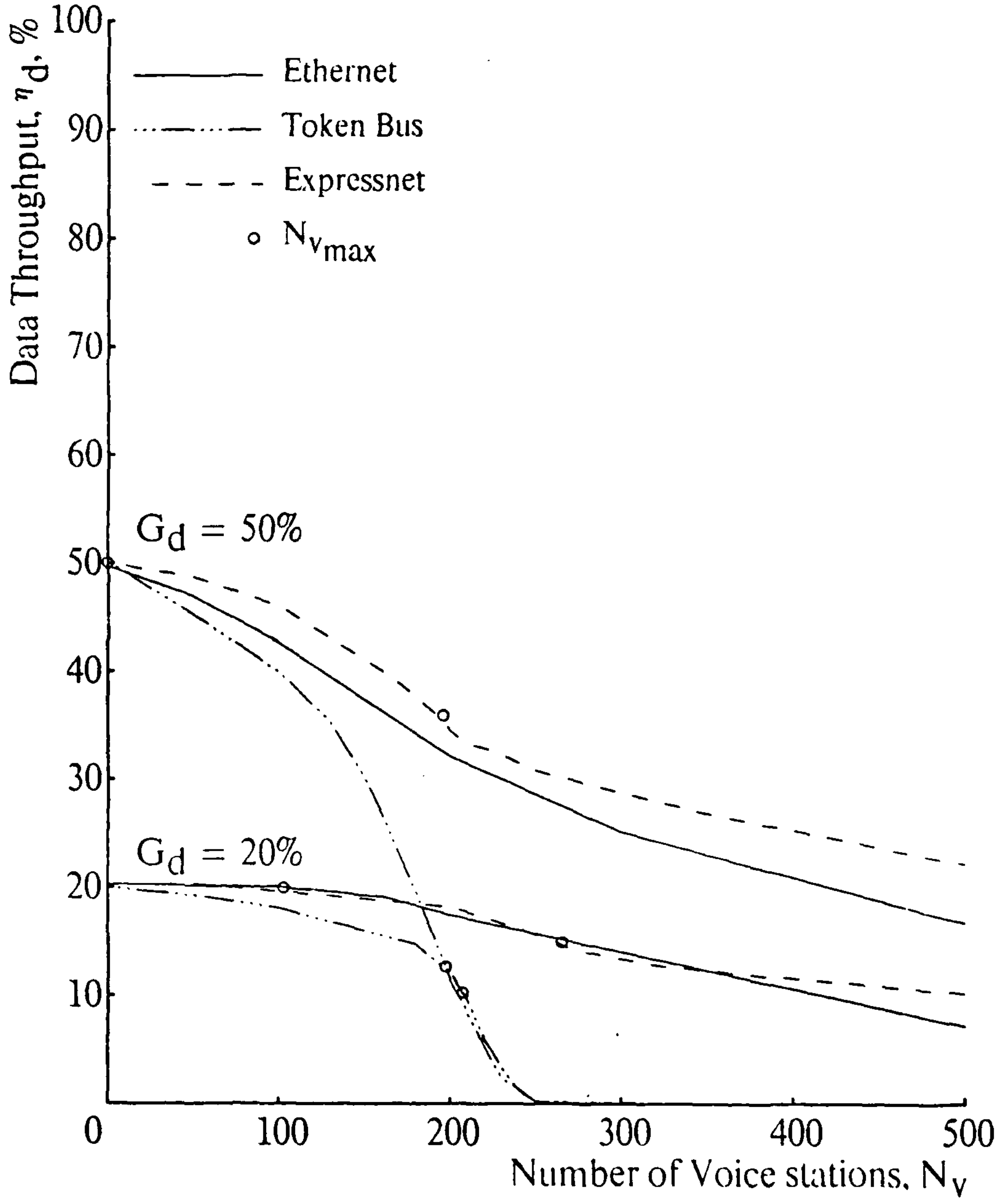


Figure 7-4: 10 Mb/s, 1 km: Total Data Throughput vs. N_v
Ethernet, Token bus, Expressnet
D_{max} = 20 ms, with silence suppression

Also indicated on the curves are the points at which $N_v = N^{(1)}_{v_{max}}$. It is desirable from the point of view of voice traffic to operate to the left of this point on each curve. In a system operating at $N_v = N_{v_{max}}$, the Ethernet achieves higher data throughput than the two round-robin schemes.

C = 100 Mb/s

The variation of data throughput with the number of voice stations at 100 Mb/s is similar to that at 10 Mb/s in the case of the Token Bus and Expressnet (Figure 7-5). With $N_v = N^{(1)}_{v_{max}}$, η_d is about 70-80% of G_d in the case of the Expressnet, but is close to zero in the Token Bus owing to the values of the priority parameters used. Choosing TRT to yield the same data throughput at $N^{(1)}_{v_{max}}$ as in the Expressnet would reduce the voice capacity of the Token Bus by 5-15%. In the Token Bus, there is a region with N_v about 1000 in which η_d is lower when G_d = 50% than when G_d = 20%. This behaviour occurs because we use a larger number of data stations to generate the higher offered load. This results in a longer period being spent in passing the token and consequently the round length exceeds TRT for smaller values of N_v. Similar behaviour, though less pronounced, is observed in Figure 7-4.

Delay

In Figure 7-6, interactive data delay is plotted as a function of N_v for D_{max} = 20 ms and G_d = 20%. Similar curves are obtained for other parameter values. As is to be expected from the throughput curves presented above, delay in the Token Bus increases rapidly to infinity as N_v exceeds $N^{(1)}_{v_{max}}$. Delay characteristics in the case of the Ethernet and Expressnet are similar to one another. The Ethernet achieves lower delay due to the equal priority for all packets and the low delay achieved by some packets that are transmitted with few or no collisions. The knee in the curves occurs at N_v close to $N^{(1)}_{v_{max}}$ and the value of D_v at this point is related to D_{max}. The standard deviation of D_v is much higher in the case of the Ethernet compared to the Expressnet (Figure 7-7). For $N_v > N^{(1)}_{v_{max}}$, standard deviation decreases after reaching a peak. In this region, all voice packets are of length D_{max}, variation in delay occurs only due to variation in data packet arrivals and in the number of voice stations in the talk state.

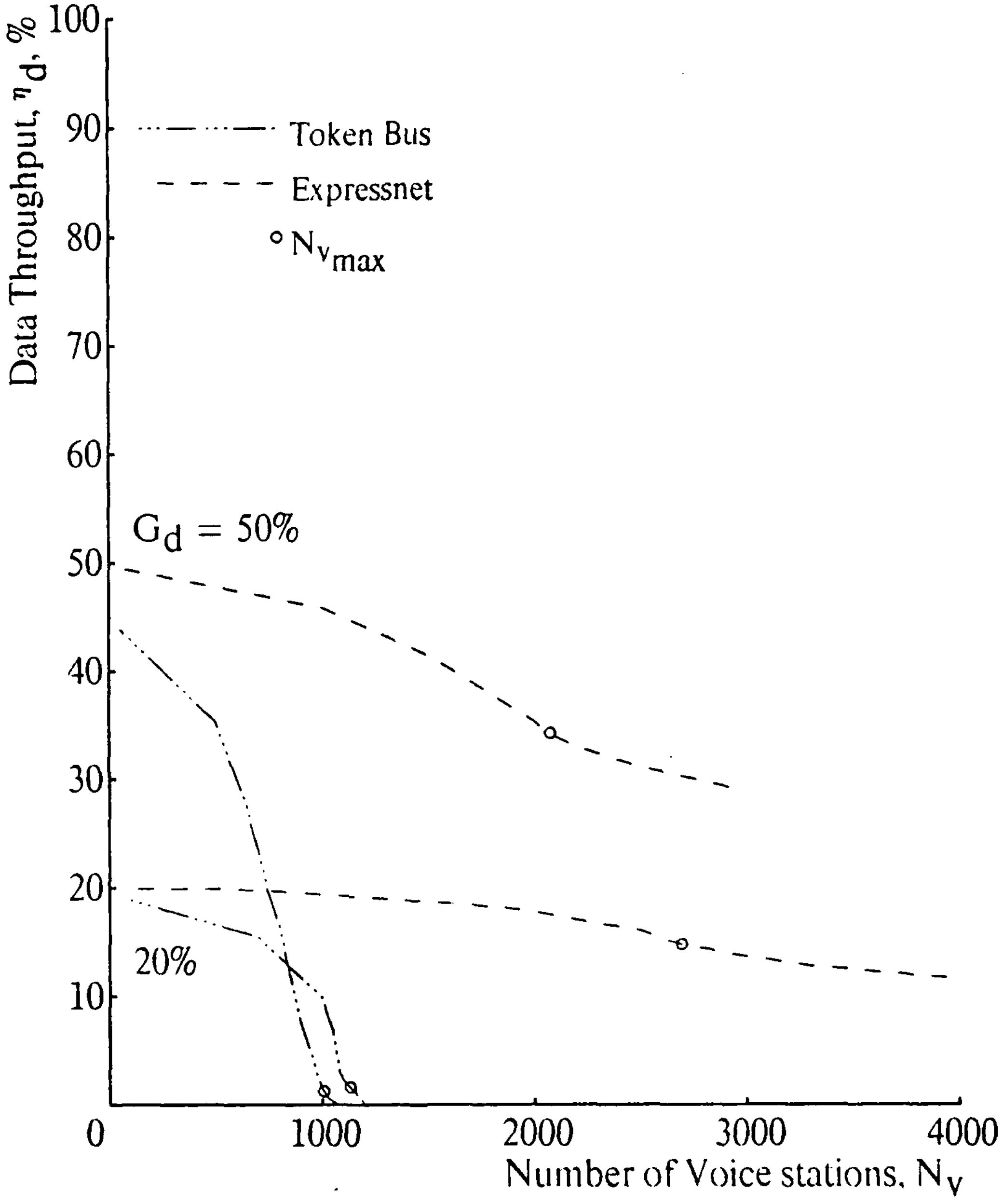


Figure 7-5: 100 Mb/s, 5 km: Total Data Throughput vs N_v.
Token Bus and Expressnet.
D_{max} = 20 ms. With silence suppression.

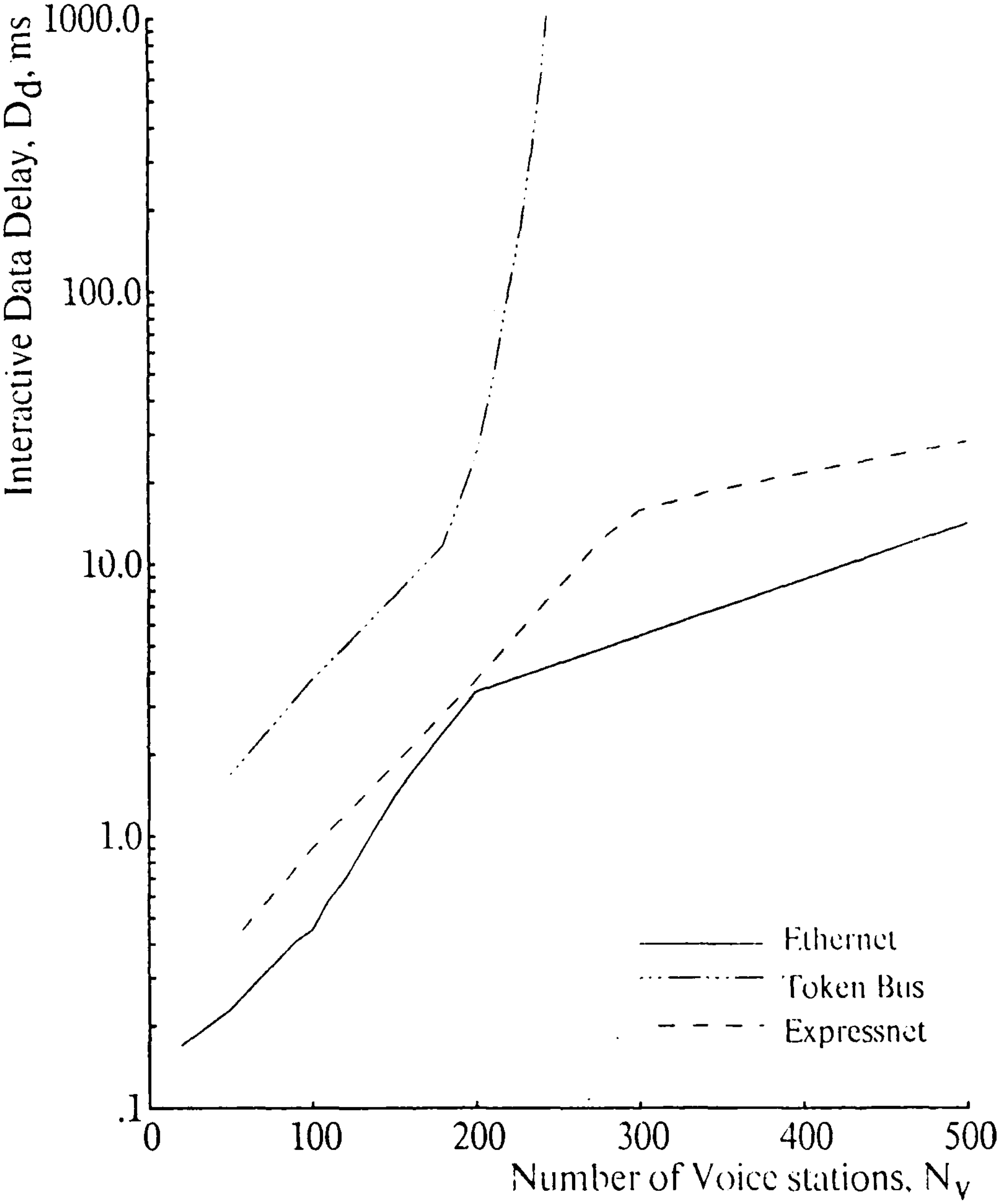


Figure 7-6: 10 Mb/s, 1 km: Interactive Data Delay vs. N_v
Ethernet, Token bus, Expressnet
$G_d = 20\%$, $D_{max} = 20$ ms, with silence suppression

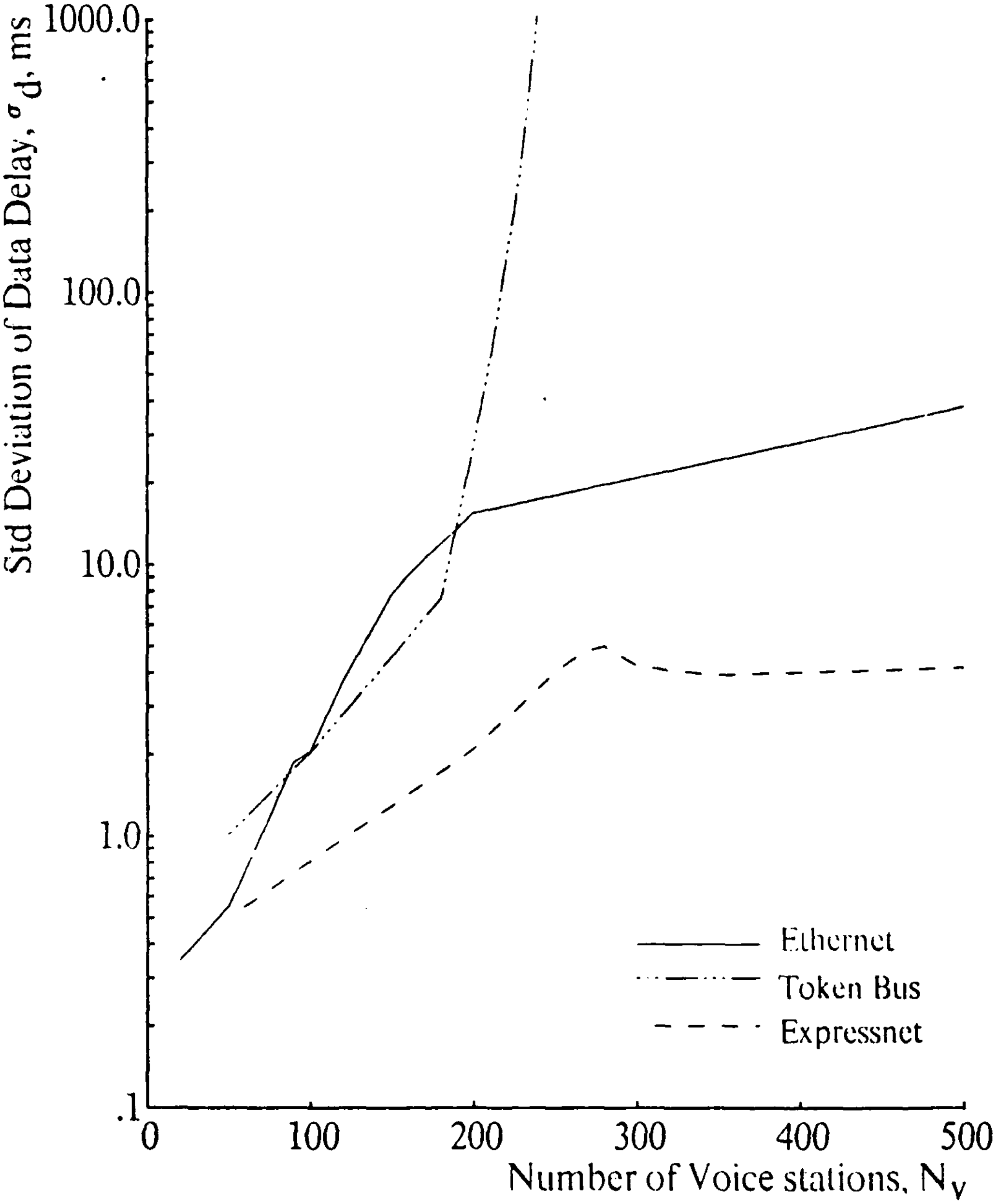


Figure 7-7: 10 Mb/s, 1 km: Std. Deviation of Interactive Data Delay vs. N_v
Ethernet, Token bus, Expressnet
G_d = 20%, D_{max} = 20 ms, with silence suppression

7.3. Summary

The use of a parametric simulator has enabled us to provide a detailed and accurate characterization of the performance of a range of broadcast bus protocols with integrated voice/data traffic. By varying key system parameters, we have covered a large volume of the design space. This study provides insights into the relative merits of random access and ordered round-robin access and of two priority mechanisms for round-robin schemes.

The contention-based Ethernet protocol is found to provide good performance at low loads and when propagation delay is relatively low (i.e., the parameter $a = \tau_p/T_p$ is small). Under these conditions, the overhead involved in the round-robin schemes results in higher delays, especially in the Token Bus with its explicit token. Under heavy loads or when a is large, however, the Ethernet is inefficient due to contention while the round-robin schemes operate in a more deterministic and hence efficient fashion, with the Expressnet with its implicit token and optimum ordering providing better performance than the Token Bus with random ordering. We note that under the assumption of optimum ordering, the performance characteristics of the Token Bus are very similar to those of the Expressnet.

At a bandwidth of 10 Mb/s and with a data load of 20% or less, the Ethernet is found to have good performance at D_{max} = 200 ms and acceptable performance at D_{max} = 20 ms. Thus, it could be used successfully as the basis for an intercom system, but is more limited in use with the public telephone network. The poor performance of the Ethernet stems from the high contention overhead per packet, on the order of several times the end-to-end propagation delay, which increases with shorter packets, and from the lack of prioritization. The two round-robin schemes, the Token Bus and Expressnet, are more suited to integrated voice/data applications. At 10 Mb/s, the Token Bus exhibits good to excellent performance at D_{max} = 20 and 200 ms, but is unacceptable at D_{max} = 2 ms. Owing to the propagation delay in passing the token, at 100 Mb/s, performance is poorer at D_{max} = 20 ms though still good at D_{max} = 200. Considering D_{max} = 20 ms, G_d = 20% and the use of silence suppression, at a bandwidth of 10 Mb/s, the voice capacities of the Ethernet, Token Bus and Expressnet are 100, 200 and 270 stations respectively. At 100

Mb/s, the Token Bus and Expressnet have capacities of 1100 and 2700 respectively. Note that the number of telephones in a system is typically 5-10 times the number that can be simultaneously active.

A detailed examination of the nature of loss indicates that in the case of the Ethernet, clips can be as high as several tenths of a second with $\varphi = 1\%$. There is high variance in the clip length and adjacent clips are uncorrelated. In the round-robin schemes, clips are on the order of several milliseconds and occur more frequently. Variance of clip length is low and correlation of adjacent clip lengths is high. The latter form of loss, short clips occurring frequently, is more acceptable to listeners than the former.

The Ethernet provides good performance to data traffic even when the number of voice stations is well above voice capacity. The average data packet delay in the Ethernet is low but variance of delay is high. In the two round-robin schemes, the performance achieved by data traffic is dependant on the choice of the priority parameters, TRT and $L_{d_{max}}$. With these parameters chosen to yield the same data throughput at $N_{v_{max}}^{(1)}$, the two schemes exhibit similar data traffic behaviour with $N_v \leq N_{v_{max}}^{(1)}$. The value of $N_{v_{max}}^{(1)}$ is seen to be dependant on the scheduling overhead. This is especially significant at high bandwidths on the order of 100 Mb/s. In such cases optimum ordering of the station significantly increases $N_{v_{max}}^{(1)}$. This optimum ordering is inherent in the Expressnet protocol but is difficult to maintain in practise in the Token Bus.

Chapter 8
Conclusions

8.1. Conclusions

The performance of broadcast bus local area networks has received considerable attention, resulting in an understanding of many of the problems of multi-access protocols. Considering a widely used implementation, the IEEE 802.3 standard which is very similar to the Ethernet, we note that several important features are difficult to model mathematically. Likewise, with integrated voice/data traffic on broadcast bus networks, prior work has provided understanding but has limitations. In this work, we have used a range of evaluation tools to address two related aspects of broadcast bus local area network performance. The first was the performance of the Ethernet under diverse traffic conditions. The second was the performance of several broadcast bus local area networks with integrated voice/data traffic. The use of measurements and detailed simulations enabled us to obtain an accurate and realistic characterization of performance, providing new insights into the problems.

We measured performance on operational 3- and 10-Mb/s Ethernets with artificially generated traffic loads. This showed that the protocol performs well over a wide range of conditions. Throughputs greater than 75% were achieved with packet lengths greater than 64 bytes and 200 bytes on the 3- and 10-Mb/s networks respectively. Average delays were usually moderate, on the order of a few milliseconds, but individual packets occasionally suffered large delays on the order of several 100 ms. The measurements also indicated the limitations of the protocol at high bandwidths and/or with short packets, i.e., when *a*, the ratio of the end-to-end propagation delay to the packet transmission time, is large. This occurs since for each packet transmission there is a contention overhead on the order of the propagation delay while stations learn of each other's transmission attempts. Further, our measurements revealed that the performance of the Ethernet is poorer than the predictions of prior analyses of the CSMA/CD protocol. This is especially true in the region of poor performance. This, however, is precisely the region in which accurate performance assessment is important. The differences are due to several differences between the implementation and the models, principally the behaviour after a collision and the number of stations on the network.

For further exploration of the protocol in the region of marginal performance, we used a detailed simulation, validated with our measurements. It was shown that performance of the standard Ethernet algorithm degrades with large numbers of stations, on the order of several hundreds. A simple modification to the algorithm enables high throughput, close to that predicted by prior analysis, to be maintained even with large numbers of stations. Other aspects studied included the effects of the number of buffers per station. These results allowed us to determine the region of applicability of the analytic models for the prediction of Ethernet performance. A simple formula for maximum throughput [Metcalfe & Boggs 76] was shown to be a good predictor with $a < 0.01$ while a sophisticated Markovian analysis [Tobagi & Hunt 80] is useful for $a < 0.1$.

Also by the use of simulation, we have quantified the performance effects of different physical distributions of stations on an Ethernet. It was shown that with symmetric distributions of stations on a linear bus, stations at the ends achieve poorer performance compared to stations near the centre. This is an effect of the non-zero propagation delay. For a station near the centre, all stations learn of its transmission attempts within half the end-to-end propagation delay, i.e., $\tau_p/2$. For a station near the end, the corresponding vlunerable period is τ_p. With asymmetric distributions, isolated stations achieve poorer performance, while stations in large clusters obtain a higher than average throughput. For example, with 39 stations clustered at one end of a 10 Mb/s, 2 km Ethernet and 1 station at the other end, with a packet length of 40 bytes, the isolated station was found to achieve a throughput of less than 1/10th that achieved by each of the stations in the cluster.

Turning next to the use of packet-switched networks for real-time voice traffic, we have proposed a new protocol for packetization of digital voice samples. Each packet is allowed to vary in length between some minimum and maximum to adapt to changing load. Thus, low delays are obtained under low loads while high utilization is maintained under heavy loads by the minimization of per-packet overhead. While the maximum packet length is determined by the delay that users can tolerate, the minimum is a function of the network protocol and other parameters. We have empirically determined the optimum minimum length over a wide range of conditions. For round-robin protocols, the value is not critical provided it is small compared to the maximum, less than about 1/10th. For random-access protocols, on the other hand, the optimum is close to the maximum.

In Chapter 2, we examined the characteristics and requirements of voice/data traffic. By the identification of key parameters and the definition of ranges of interest for these parameters, we formulated a network-independent framework for the comparative evaluation of broadcast bus local area networks. For reasons of consistency and accuracy, we developed a parametric simulator for multiple traffic types on broadcast bus local area networks. In a systematic study of representative networks, key parameters were varied over a wide range, thus providing numerical results, via interpolation, over a large region of the design space. Networks evaluated were the random-access Ethernet (IEEE 802.3 standard) and two round-robin schemes, the Token Bus (IEEE 802.4 standard) and the experimental Expressnet.

Broadly speaking, random access schemes can provide similar performance to the round-robin schemes at light loads. Under heavy loads, however, the round-robin schemes operate in a more deterministic manner and provide better performance. The Ethernet was shown to provide satisfactory service at moderate bandwidths, up to 10 Mb/s, and when delays of 20 ms and greater can be tolerated by voice traffic. At higher bandwidths, with high data loading, or under stringent delay constraints of 2 ms, the voice performance is poor. While data traffic is able to efficiently utilize bandwidth temporarily unused by voice traffic, fluctuations in data traffic lead to loss of voice samples.

The contention-free round robin schemes provide good performance even at high bandwidths of 100 Mb/s and under 2 ms delay constraints. In the case of the Token Bus, performance at moderate bandwidths was found to be independent of the ordering of the stations in the token-passing ring since the propagation delay is negligible compared to the packet transmission time. At high bandwidths, this ordering becomes important. In the Expressnet, with inherently optimum ordering, performance was good over the entire range of interest. With a data load of 20%, a delay constraint of 20 ms, the maximum number of 64 Kb/s voice stations that can be accommodated at a bandwidth of 10 Mb/s is 100, 200 and 270 in the Ethernet, Token Bus and Expressnet respectively, assuming the use of silence suppression. At 100 Mb/s, the corresponding capacities are 1100 and 2700 voice stations for the Token Bus and Expressnet respectively. Note that the number of telephones that a system can support is typically 5-10 times the voice capacity. Thus, these broadcast bus networks can support quite large telephone systems.

We have investigated two priority mechanisms for round-robin schemes, the token rotation timer (TRT) and the alternating round mechanisms. Both mechanisms provide similar performance, with the latter being marginally superior. For a given data throughput, the alternating round mechanism allows a voice capacity up to 10% greater than that with the TRT mechanism.

Differences in the nature of the loss of voice under overload were found between the networks. In the Ethernet, clips are moderately large and highly variable with low correlation between the lengths of adjacent clips due to the random order in which competing stations achieve network access. Some clips may be several 100 ms in duration, much longer than the threshold of 50 ms [Campanella 76] at which the individual clips become subjectively perceptible rather than merely contributing to background noise. In the round-robin schemes, for the same total loss, clips are much shorter than 50 ms and are more frequent with high correlation between the lengths of adjacent clips. In these schemes, under overload, every station suffers a similar clip in every round. This is subjectively more acceptable.

In summary, by the use of appropriate evaluation tools, especially measurement and detailed simulation, we have provided new insights into the behaviour of the Ethernet protocol under diverse conditions. The use of simulation and a uniform framework for evaluation has enabled us to study the trade-offs involved in the design of access protocols and priority mechanisms in broadcast bus local area networks for integrated voice/data traffic.

8.2. Suggestions for Further Work

The broad scope of this work provides several avenues for further research. One is the inclusion of other classes of interconnection structures, such as star and circular topologies (see Chapter 1). The star topology includes the digital PBX switches traditionally used for voice telephony. For cases where high data loads are expected, the Ethernet could benefit from some form of priority. Several schemes have been proposed in the literature. A scheme such as MSTDM [Maxemchuk 82] could prove useful (see Section 1.2).

Given the large number of data points we have provided, it may be possible to develop approximate analyses for interpolation and extrapolation. While this may be relatively easy in the case of the round-robin schemes, in the Ethernet, finding an approximation that is valid over a sufficiently wide range as to be useful may prove difficult. The principal obstacle is the back-off algorithm which depends on the number of successive collisions that a packet has suffered. This necessitates a large state space even for small networks (see Section 4.3). A possible approach is to model the network as a single load-dependant server, with the service rate derived from our data.

We have held several parameters fixed in our evaluations. These include the precise mix of bulk and interactive data traffic, the number of data stations used to generate a given data load, and the arrival processes used. Our experience suggests that for realistic ranges these will not significantly alter our results. The one parameter that will have a significant effect is the voice digitization rate, V. Lower encoding rates are likely to provide an approximately linear increase in voice capacity in the round-robin schemes. In the Ethernet, with utilization dependant on packet length, lowering V beyond a point may actually lead to a decrease in voice capacity. It is also realistic to consider a mix of voice stations having differing encoding rates and differing delay constraints.

In our comparisons we have implicitly assumed that the cost of the various alternatives is the same. While this assumption is justified to some extent by the fact that the physical transmission medium can be made the same in all the protocols studied, the implementations of the network-interface units differ significantly. In the Token Bus, mechanisms to handle error conditions such as loss or duplication of the token, which we have not considered, add considerably to the complexity of the implementation. The Expressnet requires simpler mechanisms for *cold-start* and keeping the network *alive*, but requires three uni-directional taps as opposed to a single bi-directional tap in the other two networks. A complete design study must include a cost/performance study and reliability considerations.

Finally, local area networks can be considered for other traffic types, such as bursty real-time traffic resulting from process control, video and facsimile in addition to voice and data. This will be particularly attractive with the development of fibre-optic networks with bandwidths on the order of 1 Gb/s. Impetus for such networks is provided by the growing interest in wide-area *integrated services digital networks* [ISDN 86].

Appendix A
Notation

The following is a summary of the notation used in this thesis.

For a variable taking on a set of values, $\{X\}$, the following notations are used:

$\bar{X}$, X_{av}	Mean of X
σ_X	Standard deviation of X
X_{min}	Minimum of X
X_{max}	Maximum of X

The following subscripts are used to qualify variables:

d, D	Data
b, B	Data (bulk)
i, I	Data (interactive)
v, V	Voice
p, P	Packet

Note: subscripts of subscripts may be omitted when the meaning is clear from the context.

The following is a list of symbols used:

C	Channel bandwidth
D	Delay
d	Network length
G	Offered load, % of C
L	Round length
L_{max}	Round maximum
N	Number of hosts
$N_{v_{max}}^{(\varphi)}$	Maximum number of voice stations with loss $= \varphi$
P	Packet length, $P = P_p + P_o + P_d$
P_d	Packet data
P_o	Packet overhead
P_p	Packet preamble

T, t	A time period, described by the subscript
t	An instant in time.
T_{ilt}	Inter-clip time
T_p	Packet transmission time
T_φ	Clip length
TRT	Token rotation timer
V	Voice coder rate
φ	Voice sample loss, % of G_v
η	Throughput, % of C
θ	Inter-packet arrival time
τ_p	Propagation delay

Appendix B The Simulator

In this Appendix we briefly describe the structure of the simulation program, emphasizing unusual aspects of the implementation. This is followed by some details of the validation.

B.1. Program Structure

The simulatoi models the system at the level of the data-link layer [Zimmermann 80] with some aspects of the physical layer included, e.g., signal propagation delay and crude modelling of some circuit delays. Thus, most aspects of the system can be captured by the use of a discrete event-driven simulator. The handling of certain continuous-time processes such as carrier-sensing pose difficulties which are dealt with later in this section. Primarily for reasons of portability, the simulator is implemented in a widely-available general-purpose language, Pascal. The simulator has been run under the TOPS-20 and Unix operating systems. The size of the simulator for the CSMA/CD, Token Bus and Expressnet protocols is about 6000 lines, excluding comments.

The program is divided into several modules (Figure B-1).[18] The station modules contain most of the protocol functions found in stations in a real system. The network module contains certain functions, such as carrier-sensing, that can be efficiently performed given global state information available in a simulator but not in a real system.

[18] Pascal does not provide independent modules. The modules we refer to are logically related procedures and functions collected in a single, separately compiled file.

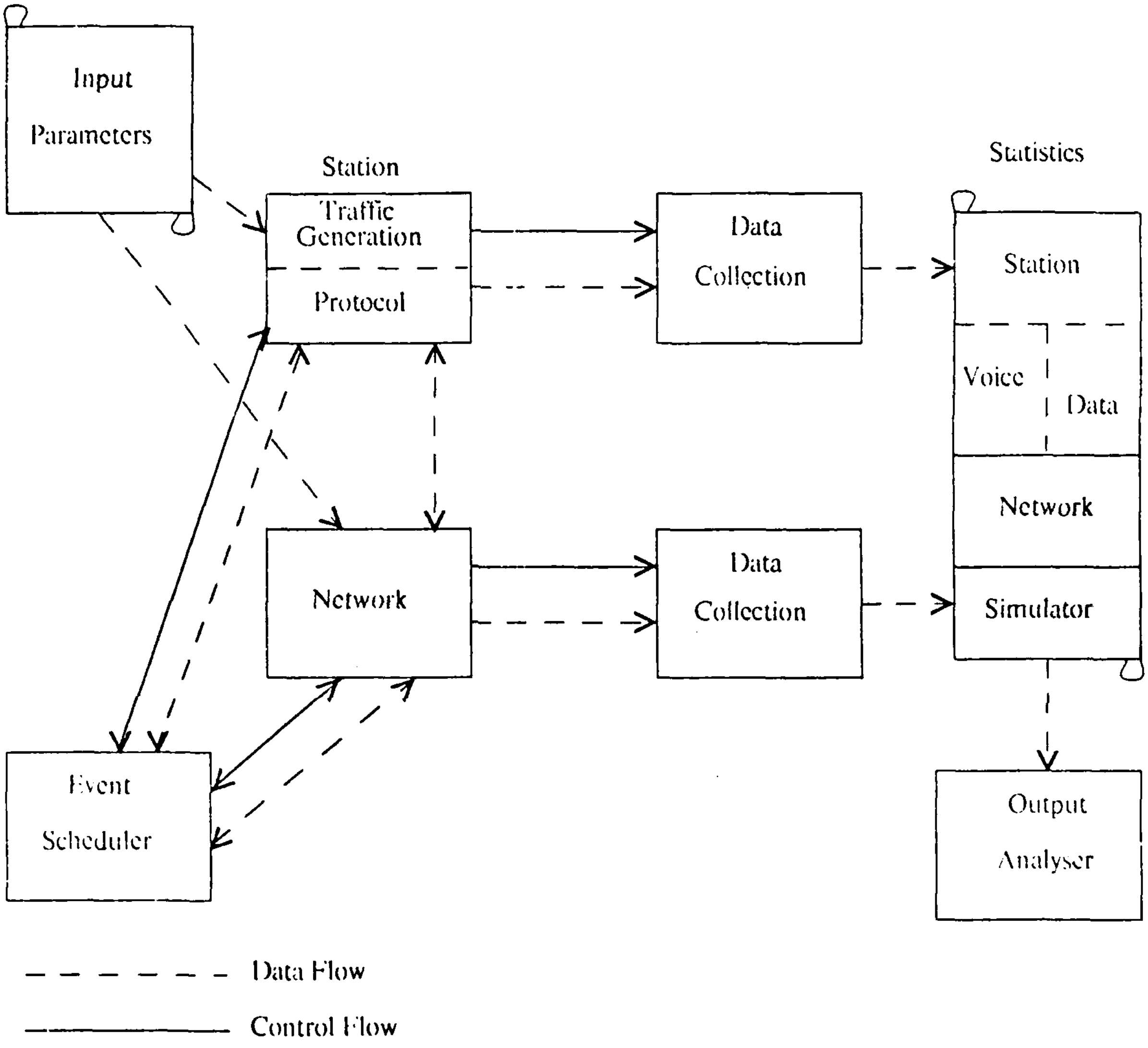


Figure B-1: Simulator Structure

Scheduler

The event scheduler maintains a list of pending events, advances the clock to the time of the next event and invokes the appropriate modules to handle the event. It also includes auxiliary procedures for servicing requests from other modules for scheduling or rescheduling of events.

In a network with a large number of stations, the number of pending events may be large. Thus, the use of a simple data structure such as a linked list for storing the events, requiring $O(n^2)$ time for performing n insert and delete operations, is inefficient. More efficient data structures exist with complexity $O(n\log n)$ for n operations. The structure we use is the *2-3 tree* (see Chapter 4 in [Aho *et. al.* 75]).

The representation of time poses a problem. In order to be able to simulate networks with widely varying bandwidths, it is desirable to use real variables for time. Pascal, however, has a limited precision for real variables, 7-8 digits for the versions used. The range from the bit-transmission time on a 100 Mb/s network, 10 ns, to the run lengths of several 10s of seconds required for accuracy of statistics easily exceeds this precision. Implementing higher precision in software is feasible but would result in greatly increased overhead in manipulation of time variables. High resolution, however, is necessary only for periods of relatively short duration, e.g., propagation delay over short segments of cable must be accurate to within nanoseconds while the run length need only be accurate to milliseconds. These can be achieved within the constraints of Pascal real variables by representing time t relative to some fixed t_0. It is merely necessary to ensure that $t - t_0 < 10^p\epsilon$, where p is the number of digits of precision available and ϵ is the desired resolution. This is accomplished by periodically incrementing t_0 by some $\delta < 10^p\epsilon$ and simultaneously decrementing all time variables by δ. With $\delta = 9$ ms, this procedure incurs an overhead of less than 1%.

Station Modules

The code for station procedures is split into two modules. The *traffic generation* module generates traffic as described in Sections 2.2.1.3 and 2.2.2.3 according to specified parameters. This is independent of the network protocol being simulated. Parameter values are specified in an input file and may be set independently for each individual station. Since the simulation is stochastic, parameters such as packet arrival rates and packet lengths have both average values and distributions specified. All stations use the same congruential multiplicative random number generation algorithm with a cycle of 2^{n-1}, where n is the word length of the computer. To reduce dependencies, this cycle is divided into sub-cycles of length 10^5 and each station uses a different sub-cycle.

The second station module is the protocol-dependant module. There is one such module for CSMA/CD protocols and one for the round-robin protocols, the Token Bus and the Expressnet. The module implements the finite-state machine shown in Figure B-2 (with some additional transitions and/or states for specific protocols).

The station is in the *idle* state while awaiting the arrival of a packet. When a packet arrives, it moves to the *queued* state, awaiting its turn in the round-robin schemes, or awaiting the end-of-carrier in CSMA. Upon either of these events, the station moves to the *trying* state where it attempts to acquire the channel. Once acquisition is successful, the station moves to the *busy* state. In this state, successful transmission of the entire packet is guaranteed. In case the acquisition attempt is unsuccessful, the station goes to the *inactive* state where it remains for some period depending on the protocol. In CSMA/CD, this corresponds to the back-off period after a collision. Depending on factors such as the number of failures, the station may decide to abandon the packet, returning to the idle state, or to retry after some period, returning to the queued state.

Network Module

The network module implements aspects of the physical layer such as signal propagation along the channel. It also includes global state information such as a list of currently transmitting stations which is used to provide to the station modules information such as whether or not carrier is present at a given location at a specified time. Thus, when the channel is busy, instead of a station sensing the channel for carrier at closely-spaced intervals to approximate the continuous process of waiting for the channel to go idle, the network module may be able to compute this from its global information. If the information cannot be computed currently, the station is placed in a queue in the network

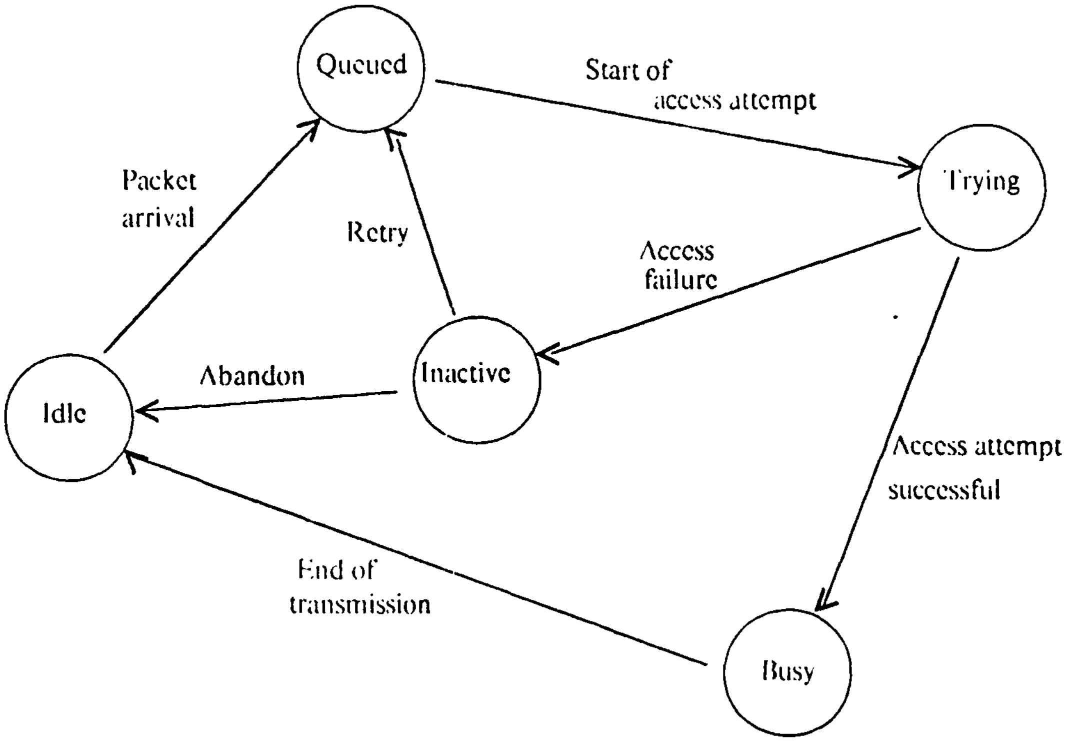


Figure B-2: Station Finite-State Machine module. For each station in this queue, when the network determines unambiguously the time that the channel will go idle at that station, it notifies the station module.

Similarly, in the round-robin protocols, it is inefficient to have the station module simulate the receiving and passing of the token for stations that do not have a packet to send. This is especially true under light loads. To avoid this, each station is added to a queue in the network module when it has a packet ready for transmission. At the end of each transmission, the network module examines its queue and uses knowledge of the order of the token-passing ring to determine which station is the next to transmit a packet with data. An appropriate event is scheduled for that station.

Input and Output

The parameters for a simulation run are specified in an input file. This file contains three sections: simulation parameters such as transient and run times; network parameters such as the protocol, bandwidth and length; and station parameters such as packet type, length, arrival process and network-interface unit parameters. All station parameters, about 10-15, may be specified independently for each station, or a common set may be specified for stations of each packet type.

The output module computes statistics of interest for each station and aggregate statistics for all stations of each packet type and for all stations. Certain aggregates, for example, average packet delay, are computed only for each packet type since it is not meaningful to average delay across packet types. Throughput, on the other hand, is computed for individual stations, for each packet type and for all stations.

B.2. Validation

We next discuss steps taken to enhance confidence in the correctness of the simulator and in the statistics obtained. The use of modular programming and the type-checking facilities afforded by Pascal were helpful in minimizing errors. Several levels of testing were used during debugging. First, tracing all events during a simulation run with a few stations and manually checking for correct operation helped eliminate several bugs. Next, for simulations with a large number of stations, the simulator was run for some time to reach steady-state. Events were then traced for some period and manually checked. This unearthed some bugs that did not occur with small numbers of stations. Finally, the simulator was run with parameters as close as possible to those in our Ethernet measurements (see Section 4.2) and various statistics were compared. Details of these are presented below on page 183. For the round-robin networks, for which exact analytic expressions are available under certain assumptions, the simulator was run under those assumptions for validation. The simulator was also run with parameters to match those used in studies in the literature. In all these tests, satisfactory correlation was obtained, with differences being attributable to minor differences in models and to statistical error.

Transient and Run Times

Since the simulator is not, in general, started under steady state conditions, it is necessary to run the simulator for some *transient period* before observations are made to allow it to reach steady state. Thereafter, observations are made for some time and various performance measures are estimated based on a finite number of samples. For example, given N observations, $\{x_1, x_2, \ldots, x_N\}$, of a random variable X, we obtain an estimate, $m = \sum_{n=1}^{N} x_n$, of the true mean, μ. To determine whether m is a good estimator of μ we obtain confidence intervals at some confidence level, typically, 95%. For this purpose, it is necessary to obtain several independent samples of m. Due to the large number of stations and the complexity of the local area networks studied, the *regenerative method* is not practical. Hence, we resort to the method of sub-runs. In the following paragraphs, we give details on the determination of suitable transient and sub-run times for use in our simulations [Kobayashi 81].

Since the random nature of the Ethernet access method is likely to result in greater fluctuations of measures with time compared to the more deterministic round-robin schemes, we determine transient and run times for the Ethernet. Several combinations of parameter values were used. We present details for a 10 Mb/s, 1 km Ethernet with data load, G_d = 20%, and the number of voice stations, N_v = 80. Since silence suppression is not used, this represents a situation of overload. The evolution of several measures with time is determined by running the simulator for run times between 0.1 and 60 s without any transient period (Table B-1). Also shown are 95% confidence intervals obtained in a benchmark run having a transient time of 20 s and run time of 100 s divided into 5 equal sub-runs. It is seen that after about 10 s most measures stabilize to within the confidence interval of the benchmark run.

		Voice Measures		Data (bulk)	
t_{run} (s)	D_{av} (ms)	φ (%)	t_{ilt} (ms)	η_d (%)	D_{av} (ms)
0.1	13.41	3.58	1.9	16.0	1.95
0.5	13.05	14.60	129.8	14.72	4.91
1.0	12.89	15.46	248.7	14.24	5.28
2.0	12.86	14.55	345.2	13.76	4.80
5.0	12.79	14.67	414.5	13.31	5.17
10.0	12.81	15.10	427.2	13.43	4.84
20.0	12.82	15.17	437.7	13.41	4.63
60.0	12.84	15.11	444.7	13.44	4.76
100.0*	12.85±0.04	15.29±0.39	445.4±7.4	13.53±0.11	4.76±0.59

* $t_{transient}$ = 20s t_{run} = 5×20 s   95% confidence intervals.

Table B-1: 10 Mb/s. 1 km Ethernet: Transient behaviour of some measures. G_d = 20%. D_{max} = 50 ms. N_v = 80. $t_{transient}$ = 0 s. Parameter t_{run}.

In the above experiments, the computation of the measures always included the initial transient period, thus biasing the results. Therefore, we conduct an alternate series of experiments with a variable transient period during which no observations are made followed by a fixed run time. Using this method, the transient behaviour is shown in Table B-2 with transient times of 0.1 to 60 s and a run time of 10 s. By eliminating the initial period during which the simulator is far from the steady state, the required transient time is much lower than in the previous method, with most measures stabilizing after a transient of about 1 s.

By similar experimentation, we arrive at suitable transient and run times for various sets of parameter values. We note that the determining factor is the number of samples. The total number of packets in a simulation run is on the order of 50,000 - 500,000 and

	Voice Measures			Data (bulk)	
$t_{transient}$ (s)	D_{av} (ms)	φ (%)	t_{ilt} (ms)	η_d (%)	D_{av} (ms)
0.1	12.81	15.16	430.6	13.26	5.44
0.5	12.79	15.02	442.9	13.29	4.80
1.0	12.80	14.99	449.9	13.33	4.72
2.0	12.86	15.26	441.7	13.37	4.86
5.0	12.87	15.35	442.1	13.58	4.42
10.0	12.84	15.23	448.1	13.39	4.44
20.0	12.80	14.92	445.0	13.17	4.81
60.0	12.85	15.32	441.1	13.43	4.74
100.0*	12.85±0.04	15.29±0.39	445.4±7.4	13.53±0.11	4.76±0.59

* $t_{transient}$ = 20s, t_{run} = 5×20 s, 95% confidence intervals.

Table B-2: 10 Mb/s, 1 km Ethernet: Transient behaviour of some measures. G_d = 20%. D_{max} = 50 ms. N_v = 80. t_{run} = 10 s. Parameter $t_{transient}$.

depends on the number of stations, the packet length and the talkspurt/silence lengths. Thus, for 10 Mb/s and D_{max} = 2 and 20 ms we use $t_{transient}$ = 5 - 10 s and t_{run} = 30 - 100 s, divided into 5 - 10 subruns. For D_{max} = 200 ms, voice packets are longer and hence we increase the times by a factor of 2 - 4. At 100 Mb/s, the number of stations is larger and we use $t_{transient}$ = 1 - 5 s and t_{run} = 5 - 20 s.

Comparison with Measurement

Comparison of simulation results with measurements on an actual system serves two purposes, namely to ensure that the simulation model is a faithful representation of the system, and to enhance confidence in the correctness of the program. In the case of the Ethernet, this is particularly important since accurate analytic models that consider all aspects of the implementation are not available (see Section 3.1). Hence, we use our measurements described in Section 4.2 for validation. Because of the nature of the implementation of the 10 Mb/s Ethernet and the stations used in our measurements, interface circuit delays could not be estimated accurately. In the 3 Mb/s Ethernet, on the other hand, the simplicity of the stations and access to logic diagrams and microcode enabled us to estimate delays with greater accuracy [Boggs 82]. Here too there are some residual differences between circuit and propagation delays in the Ethernet and in the simulation. Thus, we do not expect exact correspondence, and will show that some modifications to the simulation to approximately model these delays improves the correspondence. In this section, we present comparisons of several performance measures obtained from simulation and measurement for the 3 Mb/s setup described in Section 4.1.2. We note that comparison of delay and throughput measures for the 10 Mb/s Ethernet yielded good correlation.

We consider the shortest packet length for which we have measurements, 64 bytes. In the absence of better information, we assume that stations are uniformly distributed on the network. Recall that we have assumed constant values for circuit delays such as carrier-detection and jam times. In reality, these values vary between stations. Further, stations are connected to the common bus via *drop cables* of varying lengths which introduce additional delays. To compensate for these delays, we introduce into the simulation some random *jitter*, t_j, in the carrier detection time which is now defined to be $t_{cd} + t_j$, where t_j is uniformly distributed in the range $[0, t_{j_{max}})$.

In Figure B-3, throughput is plotted as a function of offered load, G, with $t_{j_{max}} = 0$ and 2 μs. Also shown is the corresponding curve from measurement. We see that without the jitter, the simulation underestimates throughput, while with $t_{j_{max}} = 2$ μs, correspondence is very close. This occurs because increasing jitter spreads out the times at which several backlogged stations attempt to transmit after the end of carrier. Thus, there is a higher probability that the signal from the first station to transmit will propagate to the others before they begin to transmit, reducing the collision rate. In Figure B-4, average packet delay is plotted as a function of G under the same conditions. We see that the increased throughput with increased jitter results in a slight increase in packet delay.

In Figure B-5, cumulative collision histograms are plotted for the conditions described above with $G = 320\%$. The height of the i^{th} bin gives the number of packets transmitted with fewer than i collisions, expressed as a percentage of the total number of successfully transmitted packets. Note that as jitter is increased, a larger fraction of packets are successful after fewer collisions and the histograms from simulation are closer to those from measurement.

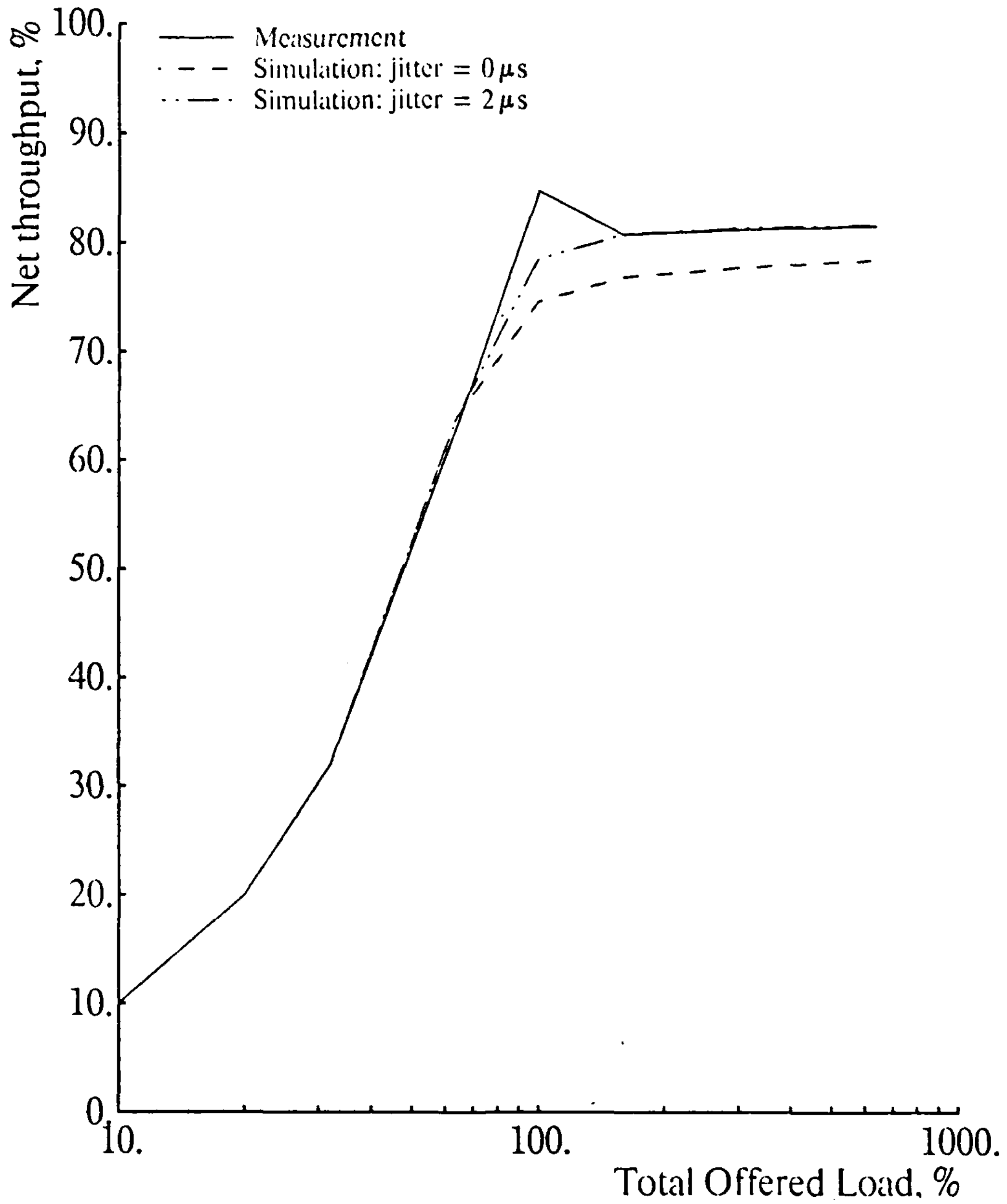


Figure B-3: 3 Mb/s, 0.55 km Ethernet: Throughput vs *G*. Measurement and simulation (with variable jitter). *P* = 64 bytes.

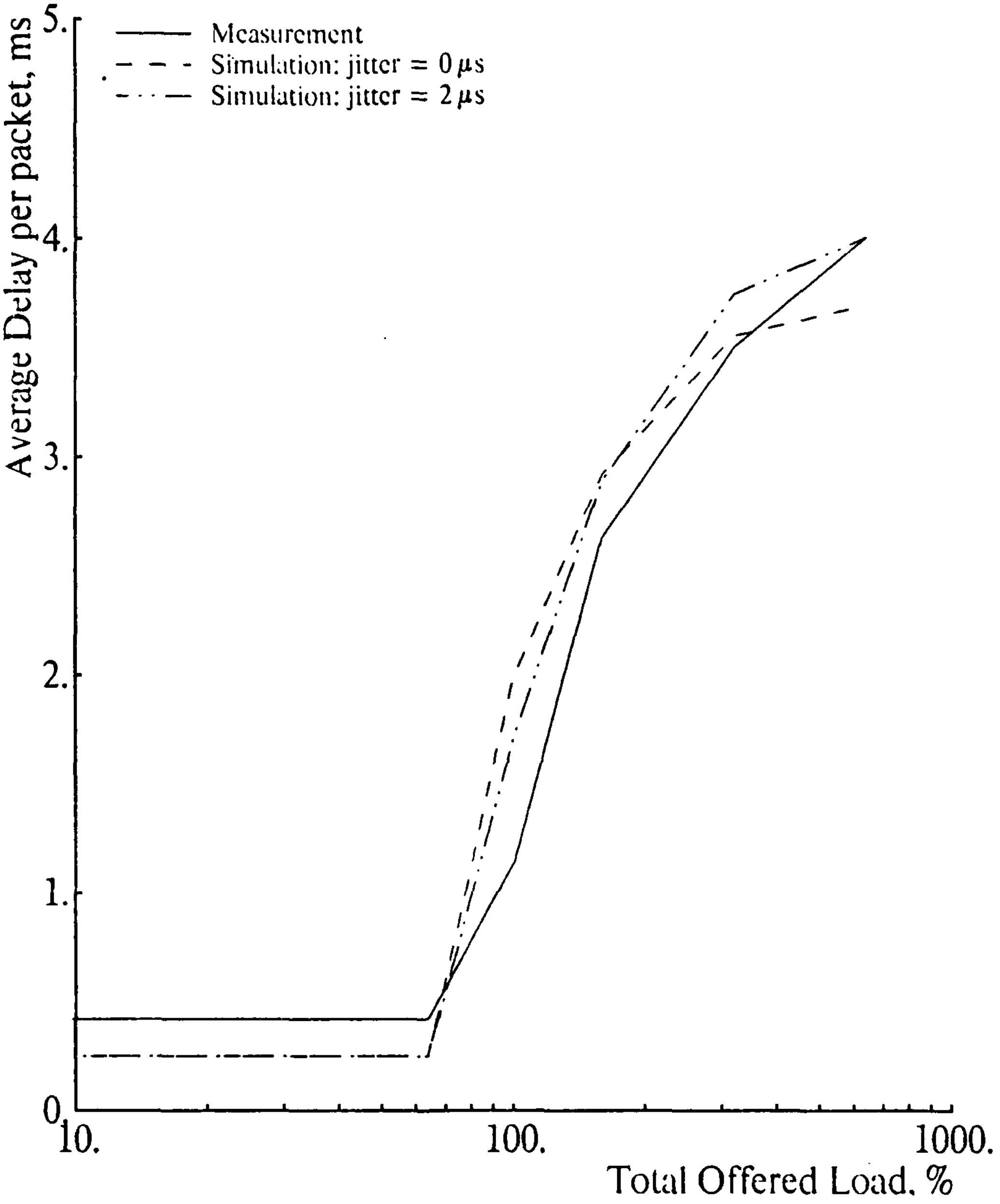


Figure B-4: 3 Mb/s, 0.55 km Ethernet: Delay vs. *G*. Measurement and simulation (with variable jitter). *P* = 64 bytes.

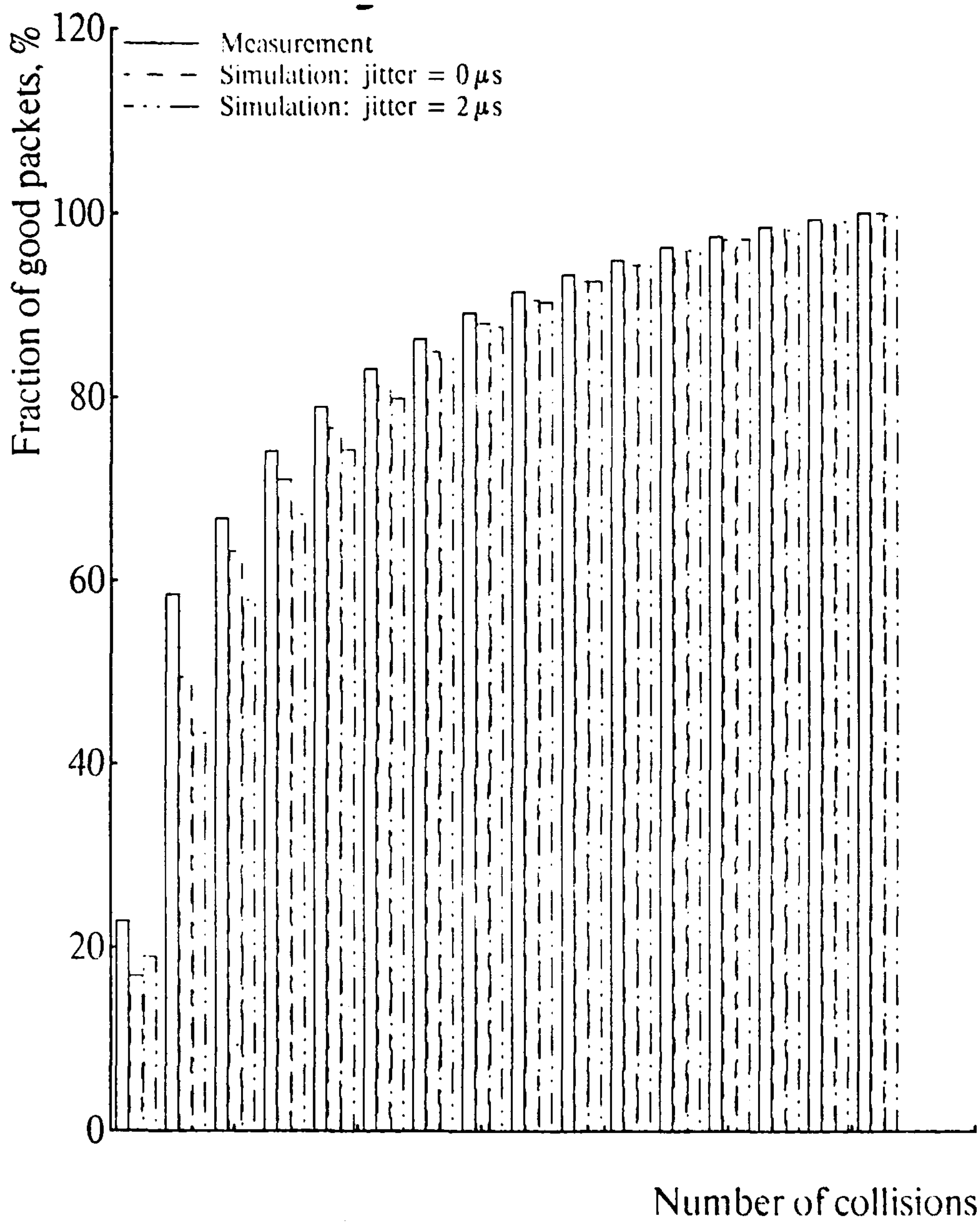


Figure B-5: 3 Mb/s, 0.55 km Ethernet: Cumulative collision histograms. Measurement and simulation (with variable jitter). P = 64 bytes. G = 320%.

In summary, without jitter, the simulation overestimates the collision rate and consequently underestimates performance compared to measurement. By the introduction of some jitter, we are able to reduce the collision rate in the simulation sufficiently that performance is slightly overestimated. The fact that the correspondence differs for different measures may be attributed to the fact that the jitter is only an approximation to some of the variable delays in the real system. Given these residual differences in delays, the correlation between measurement and simulation may be said to be good.

References

[Abramson 70] N. Abramson.
The ALOHA System - Another Alternative for Computer Communications.
In *AFIPS Conf. Proc., 1970 Fall Joint Computer Conf.*, pages 281-285. 1970.

[Aho *et. al.* 75] A.V. Aho, J.E. Hopcroft, & J.D. Ullman.
The Design and Analysis of Computer Algorithms.
Addison-Wesley Publishing Company, 1975.

[Almes & Lazowska 79]
G. T. Almes, and E. D. Lazowska.
The Behaviour of Ethernet-like Computer Communication Networks.
In *Proc. of 7th Symp. on Operating Sys. Prins., Asilomar, California*, pages 66-81. December, 1979.

[Anderson & Jensen 75]
G.A. Anderson & E.D. Jensen.
Computer Interconnection Structures: Taxonomy, Characteristics, and Examples.
ACM Computing Surveys 7(4):197-213, December, 1975.

[Arthurs & Stuck 79]
E. Arthurs & B.W. Stuck.
A Theoretical Traffic Performance Analysis of an Integrated Voice-Data Virtual Circuit Packet Switch.
IEEE Transactions on Communications COM-27(7):1104-1111, July, 1979.

[Bellamy 82] J.C. Bellamy.
Digital Telephony.
John Wiley & Sons, 1982.

[Bially *et. al.* 80] T. Bially, A. J. McLaughlin, and C. J. Weinstein.
Voice Communication in Integrated Digital Voice and Data Networks.
IEEE Transactions on Communications COM-28(9):1478-1490, September, 1980.

[Boggs 82] D. R. Boggs.
Private communication, Xerox PARC, 1982.
1982

[Boggs *et. al.* 80] D. Boggs, J. Shoch, E. Taft, and R. Metcalfe.
Pup: An Internetwork Architecture.
IEEE Transactions on Communications COM-28(4):612-624, April, 1980.

[Brady 68] P. T. Brady.
A Statistical Analysis of On-Off Patterns in 16 Conversations.
Bell System Technical Journal 47(1):73-91, January, 1968.

[Bullington & Fraser 59]
K. Bullington & J. Fraser.
Engineering Aspects of TASI.
Bell System Technical Journal 38:, March, 1959.

[Campanella 76] S. J. Campanella.
Digital Speech Interpolation.
Comsat Technical Review 6(1):127-158, Spr., 1976.

[Chandy *et. al.* 75]
K. M. Chandy, U. Herzog, & L. Woo.
Parametric Analysis of Queuing Networks.
IBM Journal of Research and Development 19(1):36-42, January, 1975.

[Cheriton 83] D. R. Cheriton.
Local Networking and Inter-Networking in the V-System.
In *Proceedings of the 8th Data Communications Symposium*, pages 9-16. Oct. 3-6, 1983.

[Chlamtac & Eisinger 83]
I. Chlamtac & M. Eisinger.
Voice/Data Integration on Ethernet: Backoff and Priority Considerations.
Technical Report 273, Dept. of Computer Science, Technion, Israel Inst. of Technology, Haifa, Israel, May, 1983.

[Chlamtac & Eisinger 85]
I. Chlamtac & M. Eisinger.
Performance of Integrated Services (Voice/Data) CSMA/CD Networks.
In *ACM SIGMETRICS Conf. on Measurement and Modeling of Computer Systems, Austin, Texas*, pages 87-93. August, 1985.

[Clark *et. al.* 78] D. D. Clark, K. T. Pogran, & D. P. Reed.
An Introduction to Local Area Networks.
Proceedings of the IEEE 66(11):1497-1517, November, 1978.

[Coviello & Vena 75]
G. Coviello & P.A. Vena.
Integration of Circuit/Packet Swithcing in a SENET (Slotted Envelope Network) Concept.
In *National Telecommunications Conf.*, pages . December, 1975.

[Coyle & Liu 83] E.J. Coyle & B. Liu.
Finite Population CSMA/CD Networks.
IEEE Transactions on Communications COM-31(11):1247-1251, November, 1983.

[Crane & Taft 80] R. C. Crane, and E. A. Taft.
Practical Considerations in Ethernet Local Network Design.
In *13th Hawaii Intl. Conf. on System Sciences, Honolulu*, pages 166-174. January, 1980.

[DeTreville 84] J. D. DeTreville.
A Simulation-Based Comparison of Voice Transmission on CSMA/CD Networks and on Token Buses.
AT&T Bell Laboratories Technical Journal 63(1):33-55, January, 1984.

[DeTreville & Sincoskie 83]
J. DeTreville & W.D. Sincoskie.
A Distributed Experimental Communications System.
IEEE Journal on Selected Areas in Communications SAC-1(5):1070-1075, November, 1983.

[Ethernet 80] *The Ethernet, A Local Area Network: Data Link Layer and Physical layer Specifications*
Version 1 edition, DEC, Intel & Xerox Corps., 1980.

[Ferrari 78] D. Ferrari.
Computer Systems Performance Evaluation.
Prentice-Hall, Inc., Englewood Cliffs, NJ-07632, 1978.

[Fine 85] M. Fine.
Performance of Demand Assignment Multiple Access Schemes in Broadcast Bus Networks.
PhD thesis, Dept. of Electrical Engg., Stanford Univ., Stanford, CA-94305, June, 1985.

[Fine & Tobagi 84]
M. Fine, & F. A. Tobagi.
Demand Assignment Multiple Access Schemes in Broadcast Bus Local Area Networks.
IEEE Transactions on Computers C-33(12):1130-1159, December, 1984.

[Fine & Tobagi 85]
M. Fine & F. A. Tobagi.
Packet Voice on a Local Area Network with Round Robin Service.
Technical Report SEL 85-275, Computer Systems Laboratory, Stanford University, Stanford, CA-94305, April, 1985.

[Fisher & Harris 76]
M.J. Fisher & T.C. Harris.
A Model for Evaluating the Performance of an Integrated Circuit- and Packet-Switched Multiplex Structure.
IEEE Transactions on Communications COM-24(2):195-202, February, 1976.

[Fratta *et. al.* 81] L. Fratta, F. Borgonovo, and F. A. Tobagi.
The Express-Net: A Local Area Communication Network Integrating Voice and Data.
In G. Pujolle (editors), *Performance of Data Communication Systems.* pages 77-88. North Holland, Amsterdam, 1981.

[Gitman & Frank 78]
I. Gitman & H. Frank.
Economic Analysis of Integrated Voice and Data Networks: A Case Study.
Proceedings of the IEEE 66(11):1549-1570, November, 1978.

[Goel & Amer 83]
A. K. Goel, and P. D. Amer.
Performance Metrics for Bus and Token-Ring Local Area Networks.
Journal of Telecommunication Networks 2(2):187-209, Spring, 1983.

[Gold 77] B. Gold.
Digital Speech Networks.
Proceedings of the IEEE 65(11):, December, 1977.

[Gonsalves 82] T. A. Gonsalves.
Packet-Voice Communication on an Ethernet Local Network: an Experimental Study.
Technical Report SEL 230, Computer Systems Laboratory, Stanford University, Stanford, CA-94305, February, 1982.

[Gonsalves 83] T. A. Gonsalves.
Packet-Voice Communication on an Ethernet Local Network: an Experimental Study.
In *ACM SIGCOMM Symposium on Communications Architectures and Protocols, Austin, Texas,* pages 178-185. March, 1983.

[Gonsalves 85] T. A. Gonsalves.
Performance Characteristics of 2 Ethernets: an Experimental Study.
In *ACM SIGMETRICS Conf. on Measurement and Modeling of Computer Systems, Austin, Texas*, pages 78-86. August, 1985.

[Gruber 81] J. J. Gruber.
Delay Related issues in Integrated Voice and Data Networks.
IEEE Transactions on Communications COM-29(6):786-800, June, 1981.

[Gruber & Le 83] J. G. Gruber, and N. H. Le.
Performance Requirements for Integrated Voice/Data Networks.
IEEE Journal on Selected Areas in Communications SAC-1(6):981-1005, December, 1983.

[Gruber & Strawczynski 85]
J. G. Gruber, and L. Strawczynski.
Subjective Effects of Variable Delay and Speech Clipping in Dynamically Managed Voice Systems.
IEEE Transactions on Communications COM-33(8):801-808, August, 1985.

[Heidelberger & Lavenberg 84]
P. Heidelberger & S. S. Lavenberg.
Computer Performance Evaluation Methodology.
IEEE Transactions on Computers C-33(12):1195-1220, December, 1984.

[IEEE 85a] *ANSI/IEEE Std 802.3-1985 - Carrier Sense Multiple Access Method and Physical Layer Specifications*
The Institute of Electrical and Electronics Engineers, Inc., 345 East 47th Street, New York, NY 10017, USA, 1985.

[IEEE 85b] *ANSI/IEEE Std 802.4-1985 - Token-Passing Bus Access Method and Physical Layer Specifications*
The Institute of Electrical and Electronics Engineers, Inc., 345 East 47th Street, New York, NY 10017, USA, 1985.

[Iida *et. al.* 80] I. Iida, M. Ishizuka, Y. Yasuda, and M. Onoe.
Random Access Packet Switched Local Computer Network with Priority Function.
In *NTC 80, Houston, TX*, pages 37.4.1-37.4.6. December, 1980.

[ISDN 86] Richard P. Skillen (editor).
Integrated Services Digital Networks (Special Issue).
IEEE Communications Magazine 24(3), Mar 1986.

[Johnson & O'Leary 81]
D. H. Johnson, and G. C. O'Leary.
A Local Access Network for Packetized Digital Voice Communication.
IEEE Transactions on Communications COM-29(5):679-688, May, 1981.

[Kleinrock 75] L. Kleinrock.
Queueing Systems, Volume 1: Theory.
John Wiley & Sons, 1975.

[Kleinrock 76] L. Kleinrock.
Queueing Systems, Volume 2: Computer Applications.
John Wiley & Sons, 1976.

[Kleinrock & Tobagi 75]
L. Kleinrock, and F. A. Tobagi.
Packet Switching in Radio Channels: Part I -- Carrier Sense Multiple-access Modes and their Throughput-delay Characteristics.
IEEE Transactions on Communications COM-23(12):1400-1416, December, 1975.

[Kobayashi 81] H. Kobayashi.
Modelling and Analysis: An Introduction to System Performance Evaluation Methodology.
Addison-Wesley Publishing Co., 1981.

[Kume 85] Hiroshi Kume.
Ethernet Evaluation in Fuji Xerox.
Technical Report, . September, 1985.

[Kurose *et. al.* 84]J. F. Kurose, M. Schwartz, and Y. Yemini.
Multiple-Access Protocols and Time-Constrained Communication.
ACM Computing Surveys 16(1):43-70, March, 1984.

[Lam 80] S. S. Lam.
A Carrier Sense Multiple Access Protocol for Local Networks.
Computer Networks 4(1):21-32, February, 1980.

[Limb & Flamm 83]
J. O. Limb, and L. E. Flamm.
A Distributed Local Area Network Protocol for Combined Voice and Data Transmission.
IEEE Journal on Selected Areas in Communications SAC-1(5):926-934, November, 1983.

[Limb & Flores 82]
J. O. Limb, and C. Flores.
Description of Fasnet - A Unidirectional Local-Area Communications network.
Bell System Technical Journal 61(7):1413-1440, September, 1982.

[Liu *et. al.* 81] T.T. Liu, L. Li & W.R. Franta.
A Decentralized Conflict-free Protocol, GBRAM for Large Scale Local Networks.
In *Proc. Computer Networking Symp.*, pages 39-54. December, 1981.

[Maxemchuk 82] N. F. Maxemchuk.
A Variation on CSMA/CD that Yields Movable TDM Slots in Integrated Voice/Data Local Networks.
Bell System Technical Journal 61(7):1527-1550, September, 1982.

[Maxemchuk & Netravali 85]
N. F. Maxemchuk, and A. N. Netravali.
Voice and Data on a CATV Network.
IEEE Journal on Selected Areas in Communications SAC-3(2):300-311, March, 1985.

[Metcalfe & Boggs 76]
R. M. Metcalfe, and D. R. Boggs.
Ethernet: Distributed Packet Switching for Local Computer Networks.
Communications of the ACM 19(7):395-404, July, 1976.

[Musser, *et. al.* 83]
J. M. Musser, T. T. Liu, L. Li, & G. J. Boggs.
A Local Area Network as a Telephone Local Subscriber Loop.
IEEE Journal on Selected Areas in Communications SAC-1(6):1046-1054, December, 1983.

[Nutt & Bayer 82]
G. J. Nutt, and D. L. Bayer.
Performance of CSMA/CD Networks Under Combined Voice and Data Loads.
IEEE Transactions on Communications COM-30(1):6-11, January, 1982.

[Roberts 78] L. G. Roberts.
The Evolution of Packet Switching.
Proceedings of the IEEE 66(11):1307-1313, November, 1978.

[Saltzer, Reed & Clark 84]
J. H. Saltzer, D. P. Reed, and D. D. Clark.
End-To-End Arguments in System Design.
ACM Transactions on Computer Systems 2(4):277-288, November, 1984.

[Shacham & Hunt 82]

N. Shacham & V. B. Hunt.

Performance Evaluation of the CSMA/CD (1-persistent) Channel-Access Protocol in Common-Channel Local Networks.

In *Proc. of IFIP TC 6 Int'l In-Depth Symposium on Local Computer Networks, Florence, Italy*, pages 401-412. April, 1982.

[Shoch 79] J. F. Shoch.

Design and Performance of Local Computer Networks.

PhD thesis, Dept. of Computer Science, Stanfora Univ., Stanford, CA-94305, August, 1979.

[Shoch 80] J. F. Shoch.

Carrying Voice Traffic Through an Ethernet Local Network -- a General Overview.

In *IFIP WG 6.4 Workshop on Local-Area Computer Networks, Zurich*, pages . . 1980.

[Shoch & Hupp 80]

J. F. Shoch, and J. Hupp.

Measured Performance of an Ethernet Local network.

Communications of the ACM 23(12):711-721, December, 1980.

[Shur 86] D. Shur.

Performance Evaluation of Multihop Packet Radio Networks.

PhD thesis, Dept. of Electrical Engg., Stanford Univ., Stanford, CA-94305, July, 1986.

[Sohraby *et. al.* 84]

K. Sohraby, M. L. Molle, & A. N. Venetsanopoulos.

Why Analytical Models of Ethernet-like Local Networks are so Pessimistic.

In *IEEE Global Telecommunications Conference, Atlanta, Georgia*, pages 19.4.1-19.4.6. November, 1984.

[Swinehart, Stewart & Ornstein 83]

D.C. Swinehart, L.C. Stewart & S.M. Ornstein.

Adding Voice to an Office Computer Network.

In *IEEE GlobeCom '83*, pages . November, 1983.

Xerox PARC Tech Rep CSL-83-8, Feb 84, 16 pp.

[Tasaka 86] S. Tasaka.

Performance Analysis of Multiple Access Protocols.

The MIT Press, Cambridge, Massachusetts, 1986.

[Thacker, *et. al* 82]
C. P. Thacker, E. M. McCreight, B. W. Lampson, and D. R. Boggs.
Alto: A Personal Computer.
In D. P. Siewiorek, C. G. Bell, and A. Newell (editors). *Computer Structures: Principles and Examples*, pages 549-572. McGraw-Hill Book Co., 1982.

[Tobagi 80] F.A. Tobagi.
Multiaccess Protocols in Packet Communication Systems.
IEEE Transactions on Communications COM-28(4):468-488, April, 1980.

[Tobagi 82] F.A. Tobagi.
Carrier Sense Multiple Access with Message-Based Priority Functions.
IEEE Transactions on Communications COM-30(1):185-200, January, 1982.

[Tobagi & Gonzalez-Cawley 82]
F. A. Tobagi, and N. Gonzalez-Cawley.
On CSMA-CD Local Networks and Voice Communication.
In *INFOCOM '82, Las Vegas, Nevada*, pages . Mar./Apr., 1982.

[Tobagi & Hunt 80]
F. A. Tobagi, and V. B. Hunt.
Performance Analysis of Carrier Sense Multiple Access with Collision Detection.
Computer Networks 4(5):245-259, Oct./Nov., 1980.

[Tobagi & Kleinrock 77]
F. A. Tobagi, and L. Kleinrock.
Packet Switching in Radio Channels: Part IV -- Stability Considerations and Dynamic Control in Carrier Sense Multiple Access.
IEEE Transactions on Communications COM-25(10):1103-1119, October, 1977.

[Tobagi *et. al.* 83] F.A. Tobagi, F. Borgonovo & L. Fratta.
Expressnet: A High-Performance Integrated-Services Local Area Network.
IEEE Journal on Selected Areas in Communications SAC-1(5):898-913, November, 1983.

[Toense 83] R. E. Toense.
Performance Analysis of NBSNET.
Journal of Telecommunication Networks 2(2):177-186, Spring, 1983.

[Weinstein & Forgie 83]
C. J. Weinstein, & J. W. Forgie.
Experience with Speech Communication in Packet Networks.
IEEE Journal on Selected Areas in Communications SAC-1(6):963-980, December, 1983.

[Zimmermann 80]
H. Zimmermann.
OSI Reference Model - The ISO Model of Architecture for Open Systems Interconnection.
IEEE Transactions on Communications COM-28(4):425-432, April, 1980.